Design Science Methodology

Prof. Dr. Roel Wieringa
University of Twente,
The Netherlands
www.cs.utwente.nl/~roelw

Motivation

- Computing Sciences
  - Information Systems, Software engineering, Artificial Intelligence, Human-Computer Interaction, etc.
- Computing sciences have evolved into disciplines with both
  - a design component and
  - an empirical research component
- Research methodology must be properly aligned with this
  - Design Science research methodology

Methodology accross knowledge disciplines

- Do these disciplines have the same methodology?
  - Technical science?
  - Social science?
  - Physical science?
  - Mathematics?
- Do they appreciate each other's methodology?
  - For social scientists, engineers are tinkerers with low social status
  - For technical scientists, social scientists are chatterboxes
  - For physicists, statistics is stamp collecting
  - The mathematicians provide the foundations of civilization, the others just don't understand it.

Our approach

- All research is **problem-solving**
  - Checklists for the design cycle and the empirical cycle
- Empirical research updates **theories**
- Designers use theories to **predict** the effect of their designs in a context

- Wieringa, R.J. (2014) *Design science methodology for information systems and software engineering.* Springer Verlag

Outline

1. What is design science
2. Research goals and problems
3. The design and engineering cycles
4. Theories
   - Conceptual frameworks
   - Generalizations
5. Empirical cycle
6. Scientific inference

Outline

1. **What is design science**
2. Research goals and problems
3. The design and engineering cycles
4. Theories
   - Conceptual frameworks
   - Generalizations
5. Empirical cycle
6. Scientific inference

- Design science is the **design** and **investigation** of artifacts in context

Reality check

- What research problems are you investigating?
- Design of conceptual /physical/software/social structures

Research problems in design science

To design an artifact to improve a problem context

Problems & Artifacts to investigate →

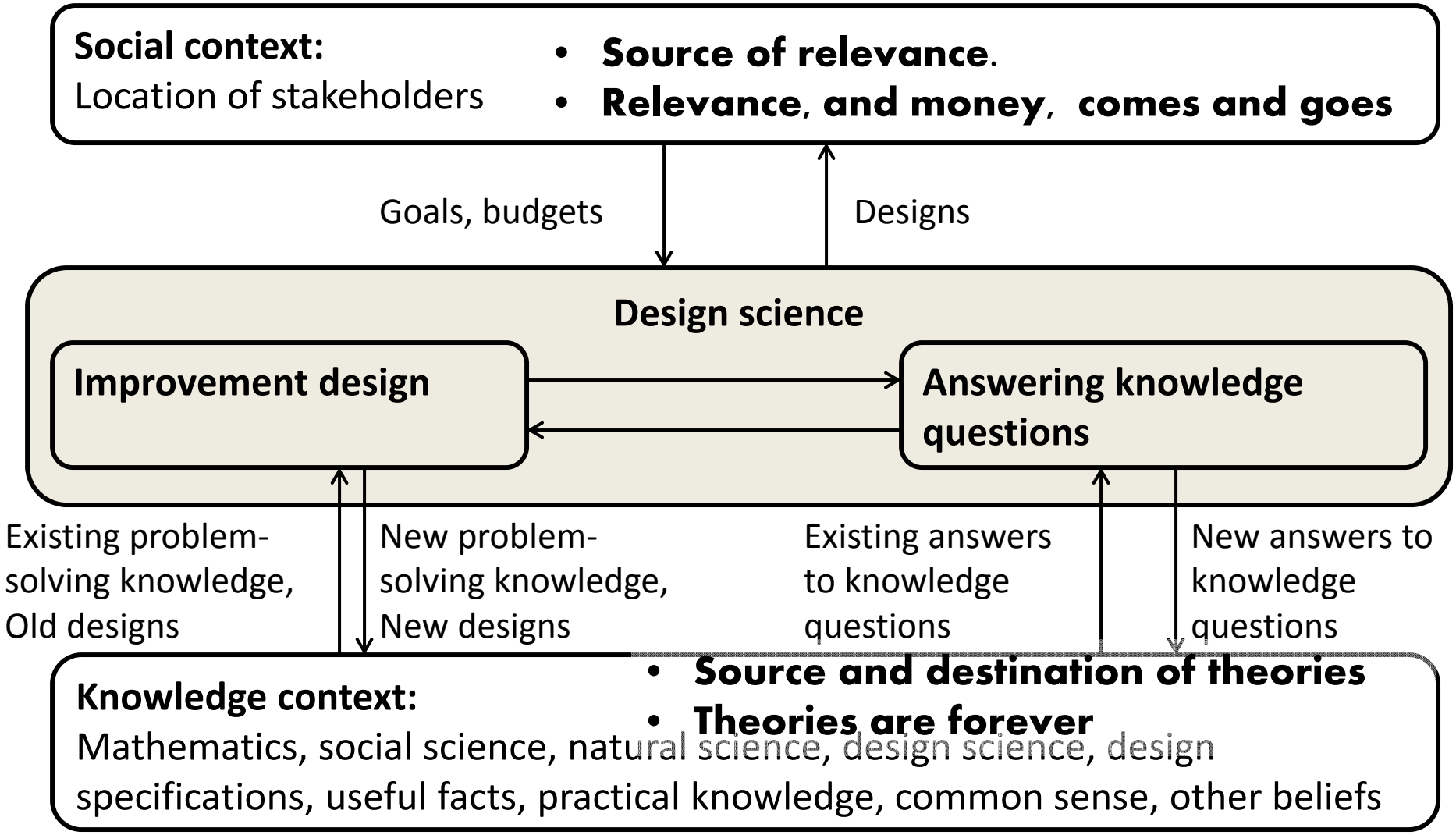
← Knowledge, Design problems

To answer knowledge questions about the artifact in context

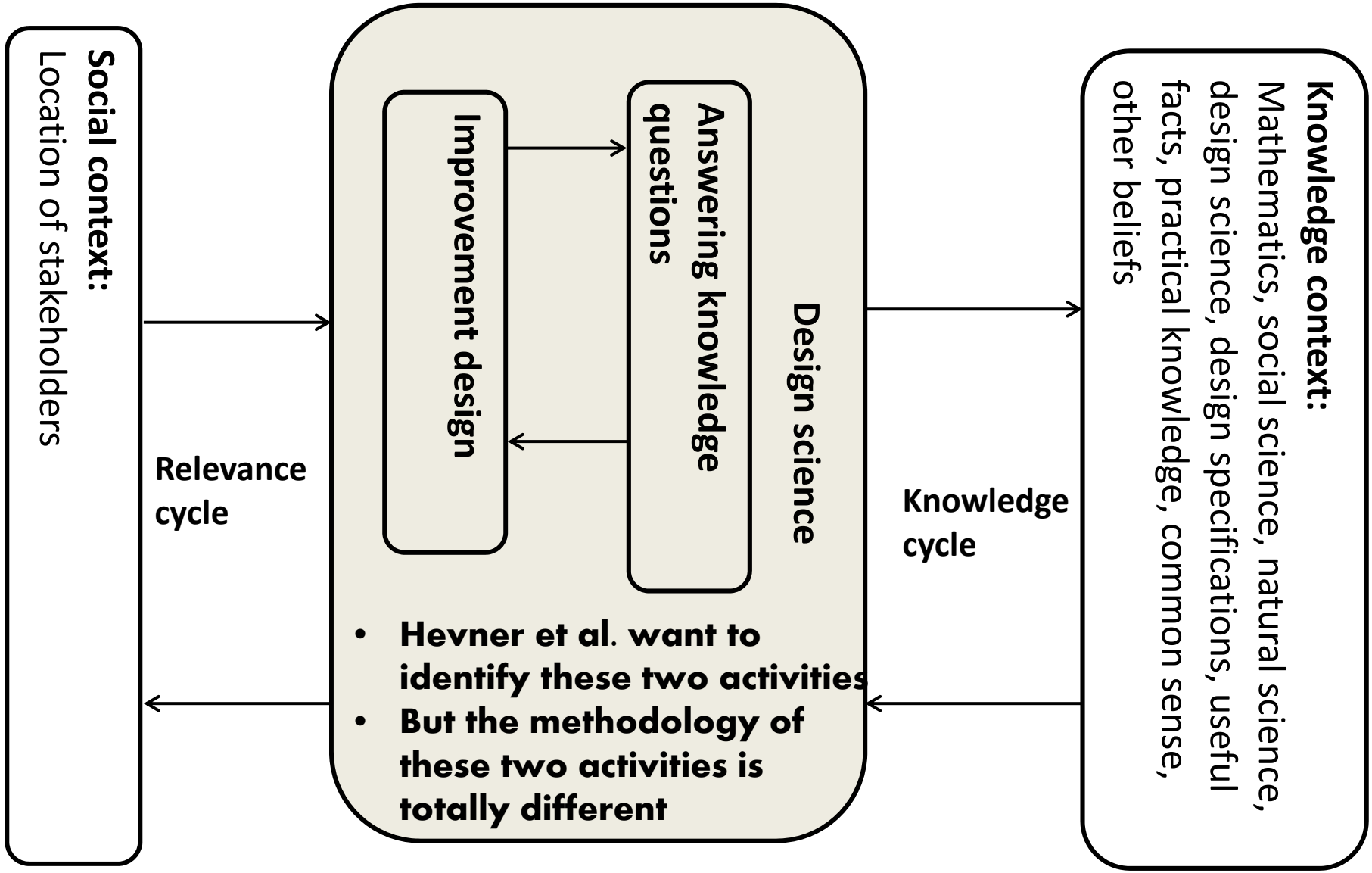
- *Design software to estimate Direction of Arrival of plane waves, to be used in satelite TV receivers in cars*
- *Design a MARP system to be used for aircraft taxi planning*
- *Design a data location regulation auditing method*
- *Design a usability and usefulness test with consultants as subjects.*

- *"Is the DoA estimation accurate enough`in this context?"*
- *Is it fast enough?*
- *"Is this agent routing algorithm deadlock-free in this context?"*
- *How much delay does it produce?*
- *Is the method usable and useful for consultants?*
- *What does this test teach me about usability?*

Framework for design science



(Dis)similarity to Hevner et al. framework



Discussion

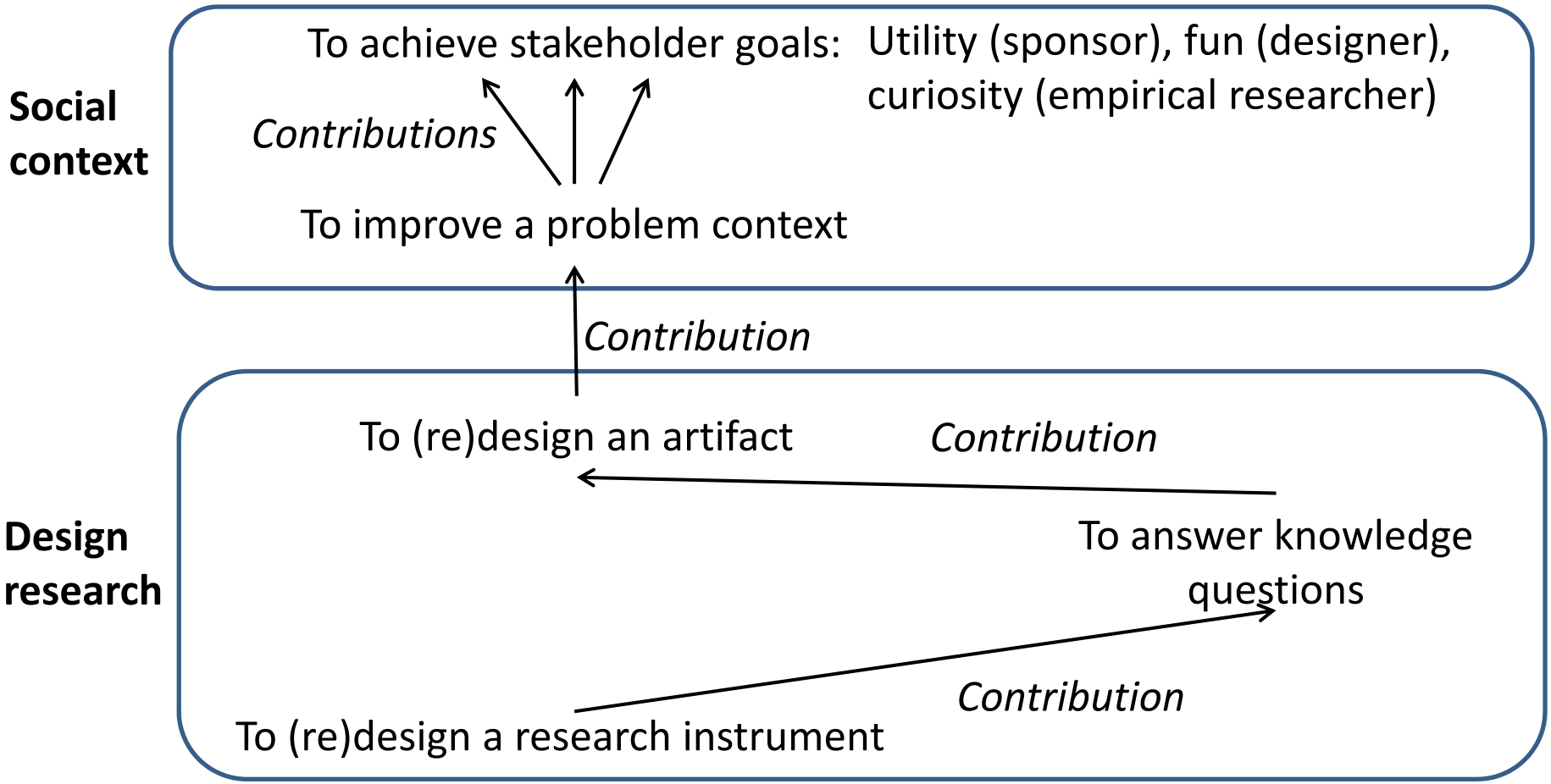
- Who are the stakeholders of your projects?
- Do they know about this?
- Who pays for the project?

Outline

1. What is design science
2. **Research goals and problems**
3. The design and engineering cycles
4. Theories
   - Conceptual frameworks
   - Generalizations
5. Empirical cycle
6. Scientific inference

2. Research Goals and Problems

Goal structure



Three kinds of design research questions

- Design problems (a.k.a. technical research questions)
  - Design <artifact> to satisfy <requirements> in order to improve <problem context> with respect to <stakeholder goals>

- Knowledge questions
  - Empirical research questions
    - Descriptive: What happened? When? Where? Who was involved?
    - Explanatory: Why?
  - Analytical research questions (math, conceptual, logical)

Reality check

- Please give examples of research questions that do not fit one of these formats.

Template for design problems

- Improve <problem context>
- by <treating it with a (re)designed artifact>
- such that <artifact requirements>
- in order to <stakeholder goals>

- *Reduce my headache*
- *by taking a medicine*
- *that reduces pain fast and is safe*
- *in order for me to get back to work*

Template for design problems

- Improve <problem context>
- by <treating it with a (re)designed artifact>
- such that <artifact requirements>
- in order to <stakeholder goals>

- ***Reduce my headache***
- *by taking a medicine*
- *that reduces pain fast and is safe*
- ***in order for me to get back to work***

Problem context and stakeholder goals.
Stakeholder language

Template for design problems

- Improve <problem context>
- by <treating it with a (re)designed artifact>
- such that <artifact requirements>
- in order to <stakeholder goals>

- *Reduce my headache*
- ***by taking a medicine***
- ***that reduces pain fast and is safe***
- *in order for me to get back to work*

Artifact and its desired properties.
Technical language

Descriptive and explanatory knowledge questions

- Descriptive questions:
  - What happened?
  - When?
  - Where?
  - What components were involved?
  - Who was involved?
  - etc.

Journalistic questions. **Yield facts**.

- Explanatory questions:
  - Why?
    1. What has **caused** the phenomena?
    2. Which **mechanisms** produced the phenomena?
    3. For what **reasons** did people do this?

Beyond the facts.

- *Descriptive question: What is the performance of this program?*
  - *Execution time for different classes of inputs*
  - *Memory usage*
  - *Accuracy*
  - *Etc. etc.*
- *Explanatory question: Why does this program have this performance?*
  1. *Cause: because it received this input (and not another input)*
  2. *Mechanism: because it has this architecture with these components*
  3. *Reasons: Because users use it only for a few simple tasks*

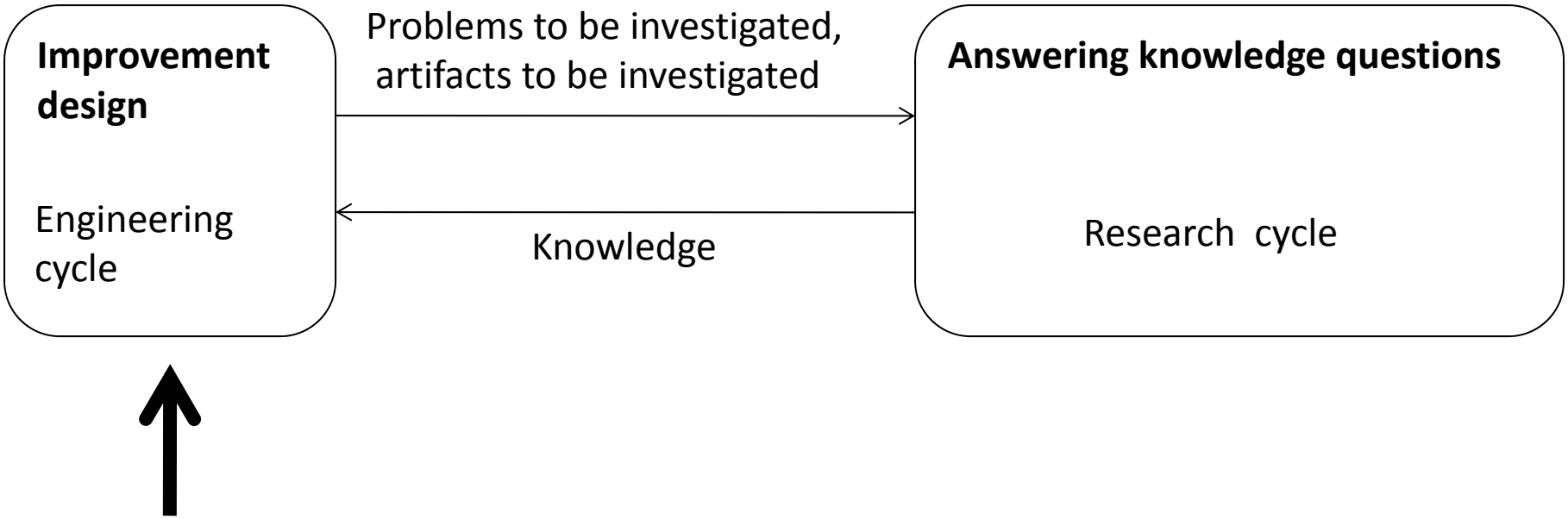
What about prediction questions?

- A prediction question is not a research problem!
  - We cannot observe the future
- To solve a prediction problem, we need a theory
  - The theory is built by answering descriptive and explanatory knowledge questions

Outline

1. What is design science
2. Research goals and problems
3. **The design and engineering cycles**
4. Theories
   - Conceptual frameworks
   - Generalizations
5. Empirical cycle
6. Scientific inference

Activities in design science

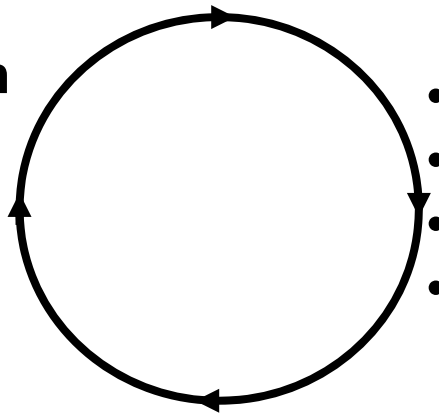


Engineering cycle

This is a checklist for what to do and how to report about it. Appendix A in the book & on my web site

! = Action
? = Knowledge question

Design implementation



Implementation evaluation = Problem investigation

- Stakeholders? Goals?
- Conceptual problem framework?
- Phenomena? Causes, mechanisms, reasons?
- Effects? Positive/negative goal contribution?

Treatment validation

- Context & Artifact → Effects?
- Effects satisfy Requirements?
- Trade-offs for different artifacts?
- Sensitivity for different Contexts?

Treatment design

- Specify requirements!
- Requirements contribute to goals?
- Available treatments?
- Design new ones!

- **Treatment** consist of introducing an artifact in the problem context
  - Treatment design = artifact design
- **Implementation** = Introduction of the treatment in the intended problem context
- We consider the case where the intended problem context is the social world, the location of stakeholders
  - So in this course implementation = transfer to the stakeholder context = technology transfer

Validation versus evaluation

- To **validate** a design is to *predict* the effects of an implementation of the design, and the utility with respect to stakeholder goals.

- To **evaluate** an implementation is to investigate the effects of an implementation *that have occurred* and their utility with respect to stakeholder goals

Design cycle

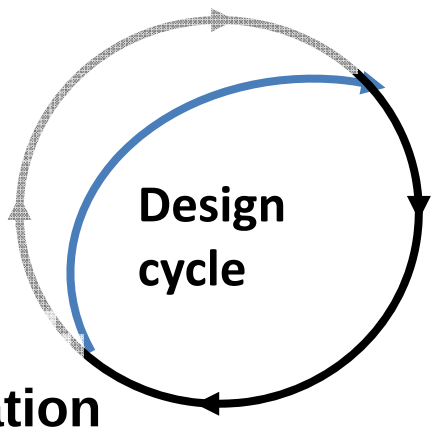
Design research projects iterate one or more times through the design cycle.

Design implementation

Implementation evaluation = Problem investigation

- Stakeholders? Goals?
- Conceptual problem framework?
- Phenomena? Causes, mechanisms, reasons?
- Effects? Positive/negative goal contribution?

This lists all your design research knowledge questions.



Treatment validation

- Context & Artifact → Effects?
- Effects satisfy Requirements?
- Trade-offs for different artifacts?
- Sensitivity for different Contexts?

Treatment design

- Specify requirements!
- Requirements contribute to goals?
- Available treatments?
- Design new ones!

This lists all your design research knowledge questions.

With which research designs can we answer them?

Research designs

	Observational study (no treatment)	Experimental study (treatment)
Case-based: investigate single cases, look at architecture and mechanisms	Observational case study	• Expert opinion, • Mechanism experiments, • Technical action research
Sample-based: investigate samples drawn from a population, look at averages and variation	Survey	• Statistical difference-making experiment

e.g. simulations

e.g. pilot project

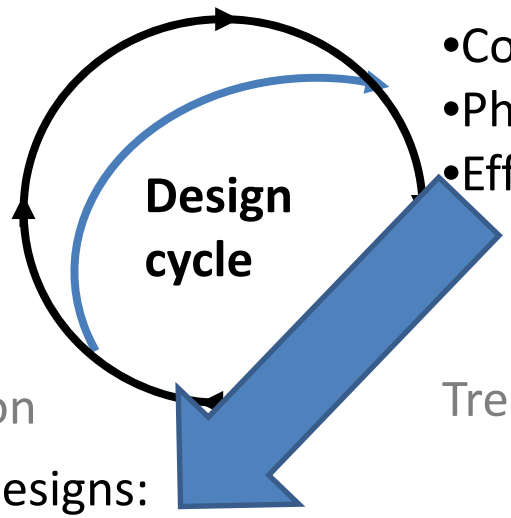
Treatment group-control group designs

Research designs in evaluation and problem investigation

Design implementation

Implementation evaluation = Problem investigation

- Stakeholders? Goals?
- Conceptual problem framework?
- Phenomena? Causes, mechanisms, reasons?
- Effects? Positive/negative goal contribution?



Treatment validation

Treatment design

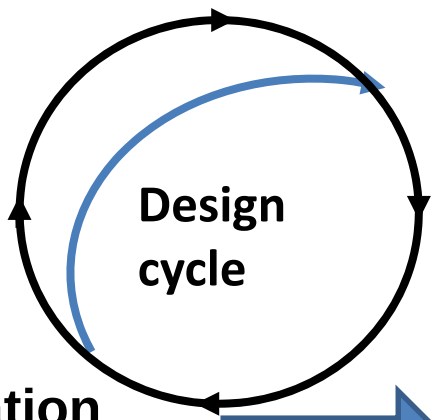
Possible research designs:

- Observational case study, *e.g. study some social network mechanisms in a global SE project*
- Survey: *e.g. survey of the use of UML in SW companies in Bra*zil
- Single-case mechanism experiment: *e.g. send messages through a company network to evaluate its performance*
- Statistical difference-making experiment: e.g. *compare two implemented techniques in two groups of professionals*

Design cycle

Design implementation

Implementation evaluation = Problem investigation

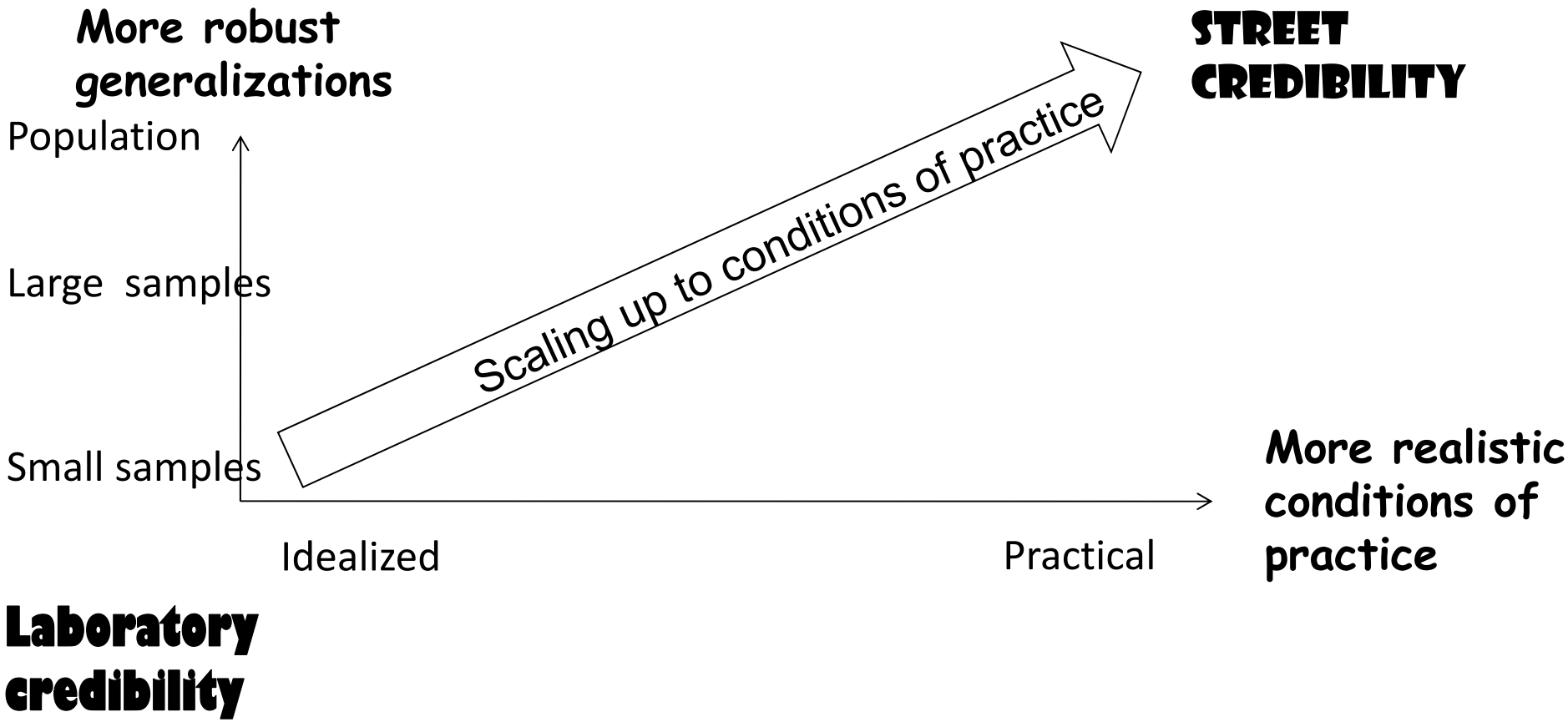


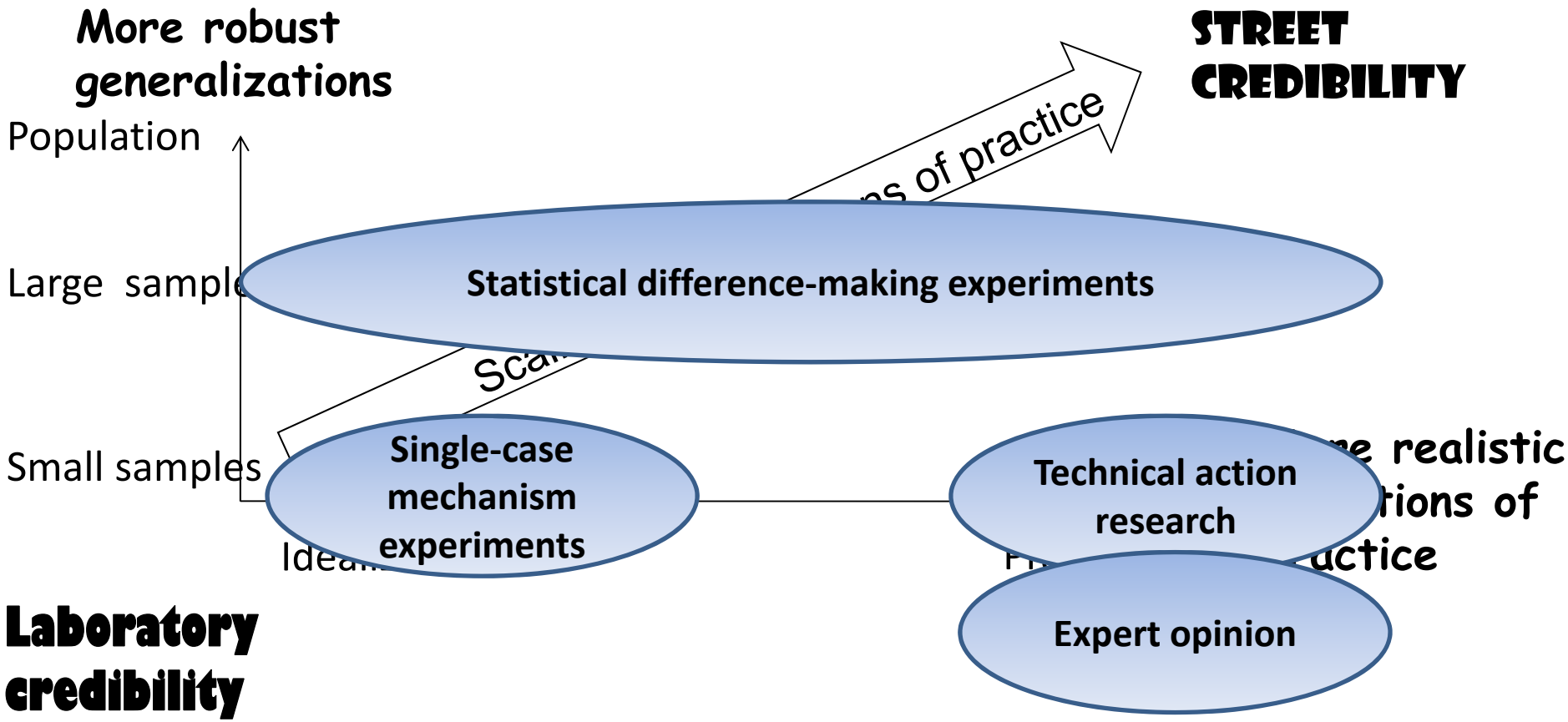
Treatment validation

•Context & Artifact → Effects?
•Effects satisfy Requirements?
•Trade-offs for different artifacts?
•Sensitivity for different Contexts?

Treatment design

- Expert opinion, *e.g. focus group*
- Single-case mechanism experiment: *test a prototype of an algorithm in simulated environment; test a prototype of a method in a classroom on students.*
- Statistical difference-making experiment: *compare two proposed techniques in two groups of students.*


More robust generalizations
STREET CREDIBILITY
Population
Large samples
Small samples
Scaling up to conditions of practice
More realistic conditions of practice
Idealized
Practical
Laboratory credibility



- Scaling up:
  - Single-case mechanism experiment (lab simulation)
  - Expert opinion
  - Single-case mechanism experiment (field simulation)
  - TAR (apply technique in a consultancy project)

Outline

1. What is design science
2. Research goals and problems
3. The design and engineering cycles
4. **Theories**
   - Conceptual frameworks
   - Generalizations
5. Empirical cycle
6. Scientific inference

- A **theory** is a belief that there is a pattern in phenomena.
  - Speculations: *“The NSA is monitoring all my email”*
  - Idealizations: *“Merging two faculties reduces cost in theory, not in practice.”*
  - Opinions: *“The Dutch lost the competition because they are not a team.’’*
  - Wishful thinking: *``My technique works better than the others.’’*
  - Scientific theories*: Theory of electromagnetism*

Scientific theories

- A **scientific** theory is a belief that there is a pattern in phenomena, that
  - Has survived tests against experience
    - Observation, measurement
    - Possibly: experiment, simulation, trials
  - Has survived criticism by critical peers
    - Anonymous peer review
    - Publication
    - Replication

- Examples
  - *Theory of electromagnetism*
  - *Theory of cognitive dissonance*
  - *The Balance theorem in social networks*
  - *Theories X, Y, Z, and W of (project) management*
  - *Technology Acceptance Model*

Theories are fallible

- Fallibilism: All theories may be wrong!
  - Outside mathematics there is no certainty
  - And even there, mathematicians make mistakes (Lakatos' *Proofs and Refutations*)
- In other words, all theories are improvable

Falsificationism

- Introduced by the philosopher Karl Popper
  - ~~Mechanical version: If the prediction of a theory is contradicted by observation, then reject the theory.~~
  - Sophisticated version: If the prediction of a theory is contradicted by observation, then
    - try to replicate it,
    - try to understand it,
    - try to improve the theory so that it can deal with the observation,
    - publish whatever comes out of this.

The structure of scientific theories

1. **Conceptual framework**
   - Definitions of concepts.
2. **Generalizations**
   - Express beliefs about patterns in phenomena.

Theory of electromagnetism

- *Conceptual framework:*
  - *Definitions of electric current, electric charge, potential difference, electric resistance, electric power, capacitance, electric field, magnetic field, magnetic flux density, inductance, …, … and their units.*
- *Generalizations*
  - *Electric charges attract or repel one another with a force inversely proportional to the square of their distance.*
  - *Magnetic attract or repel one another in a similar way and always come in north-South pairs.*
  - *An electric current inside a wire creates a corresponding circular magnetic field outside the wire.*
  - *A current is induced in a loop of wire when it is moved towards or away from a magnetic field*

Theory of cognitive dissonance

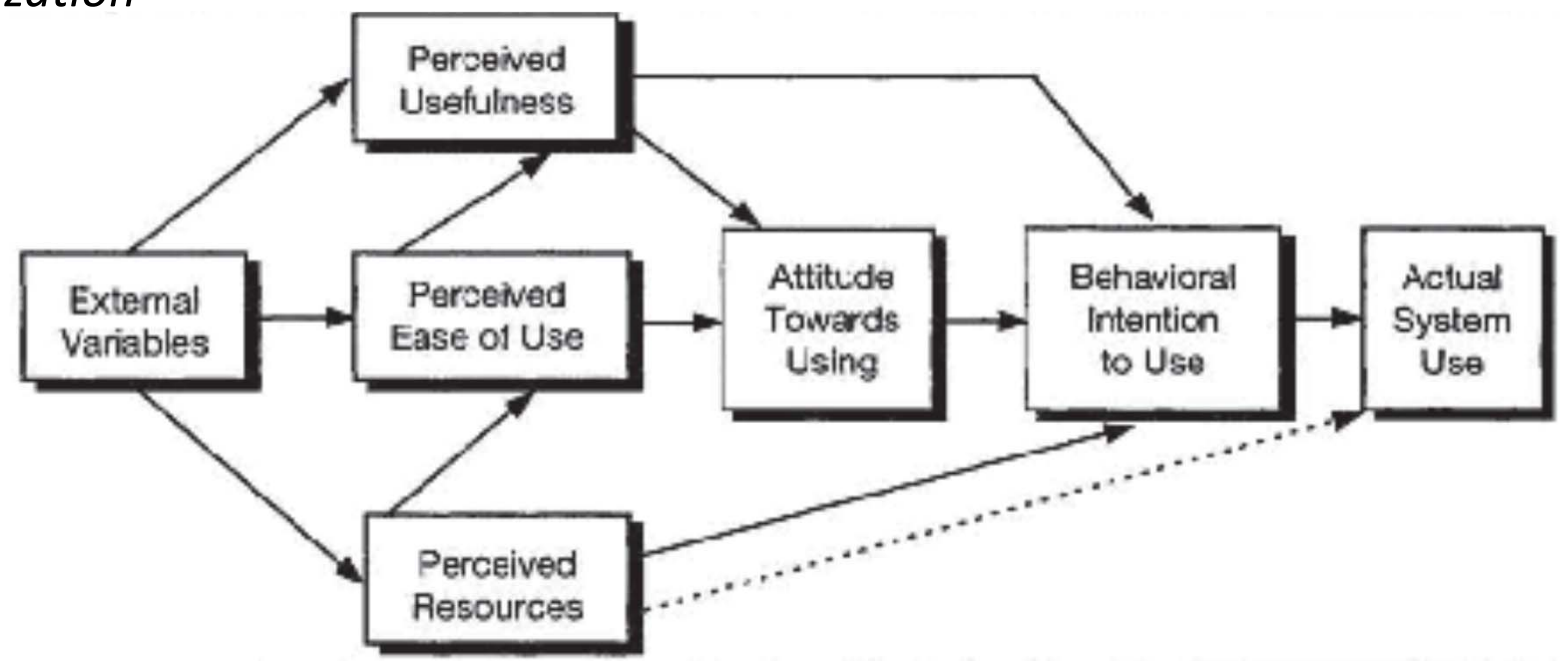
- *Conceptual framework*
  - ***Cognitive dissonance*** *is the mental stress experienced by an individual who holds contradictory beliefs or values, or is confronted by new information that conflicts with existing beliefs or values*
- *Generalization*
  - *People engage in dissonance reduction to bring their beliefs and actions in line with one another four ways:*
    - *Change behavior to agree with belief ("eat less chocolate")*
    - *Change belief to justify behavior ("occasional chocolate eating is OK")*
    - *Add new intention or behavior ("I'll work out tomorrow")*
    - *Deny information that conflicts with existing beliefs ("this is low-fat chocolate")*

The Balance Theorem in social networks

- *Conceptual framework*
  - *Definition of concepts of graph, link, friend/enemy, complete graph (each pair of nodes connected), balanced graph (no --- or ++- triangles)*
- *Generalization*
  - *Large call networks are almost balanced (V.D. Blondel, A. Decuyper and G. Krings - ``A survey of results on mobile phone datasets analysis'')*
  - *Mathematical theorem: If a labeled complete graph is balanced, then*
    - *either all pairs of nodes are friends,*
    - *or else the nodes can be divided into two groups, X and Y, such that every pair of nodes in X like each other, every pair of nodes in Y like each other, and everyone in X is the enemy of everyone in Y .*
- Idealizing assumptions **may** allow us to understand real-world phenomena

Technology Acceptance Model

- *Conceptual framework*
  - *Definitions of perceived usefulness, perceived ease of use, perceived resources, attitude towards using, behavior intention to use, actual system use*
- *Generalization*



- K. Mathieson, E. Peacock, W. W. Chin - Extending theTechnology Acceptance Model: The Influence of Perceived User Resources. SIGMIS Database, 2001.

Outline

1. What is design science
2. Research goals and problems
3. The design and engineering cycles
4. **Theories**
   - **Conceptual frameworks**
   - Generalizations
5. Empirical cycle
6. Scientific inference

- A **conceptual framework** consists of definitions of concepts , often called **constructs**.

- *Variables, relations among variables*
- *Populations, probability distribution of variable*
- *Entities, attributes, relations, taxonomy, cardinality, …*
- *Events, actions, processes, procedures, constraints , …*

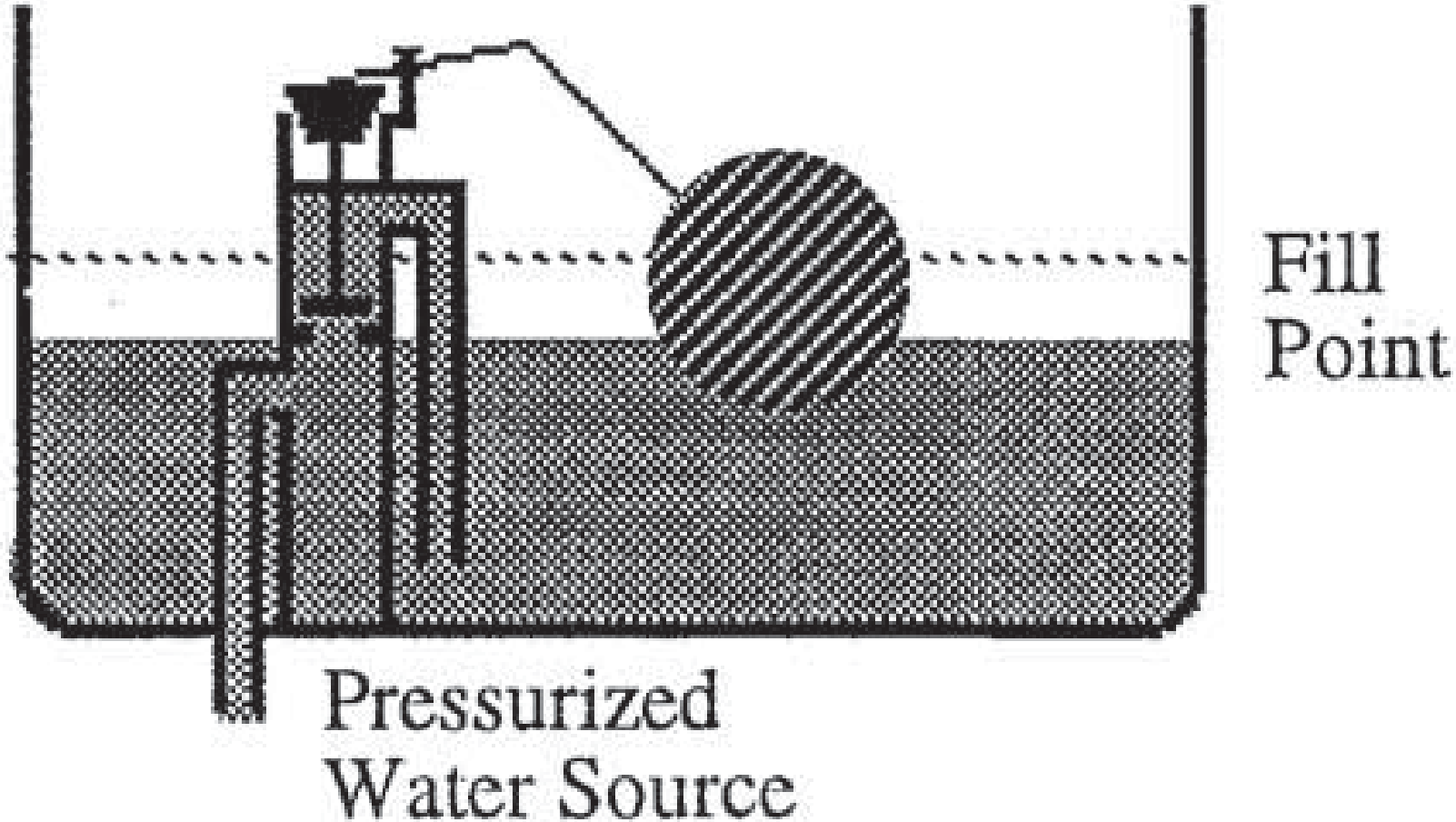
- We can define two kinds of conceptual structures

1. **Architectural structures**: Class of physical/social/technical systems, architecture, components with capabilities, interactions

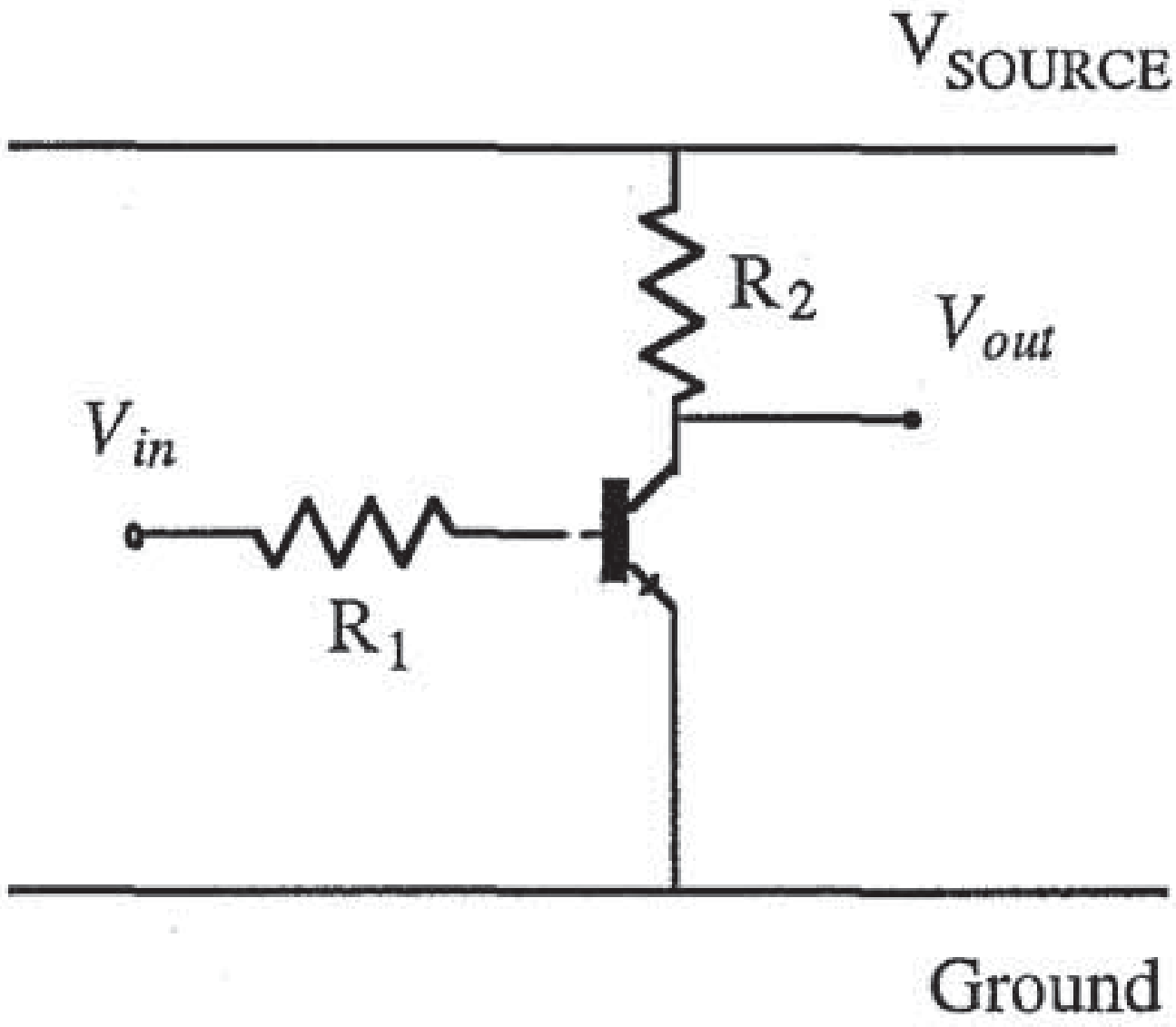
2. **Statistical structures**: Population, variables with probability distributions, relations among variables

Architectural structures

- The world is structured as a collection of systems
  - Social, physical, technical, psychological, biological, …
  - Each system has **components** with **capabilities and limitations** of behavior
  - that can **interact** with each other in specific ways.

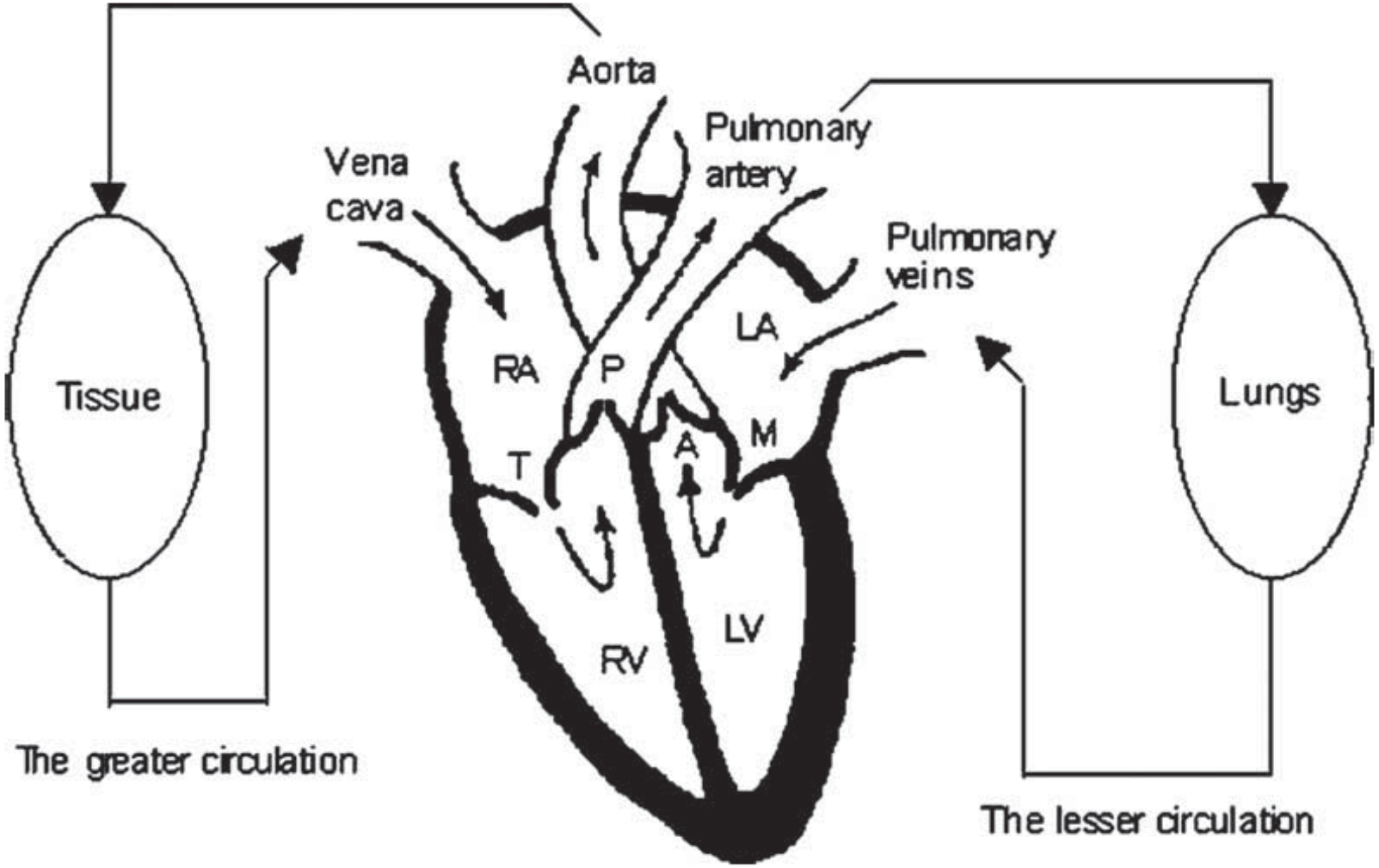


- Glennan - ``Mechanisms and the nature of causation''. 1996

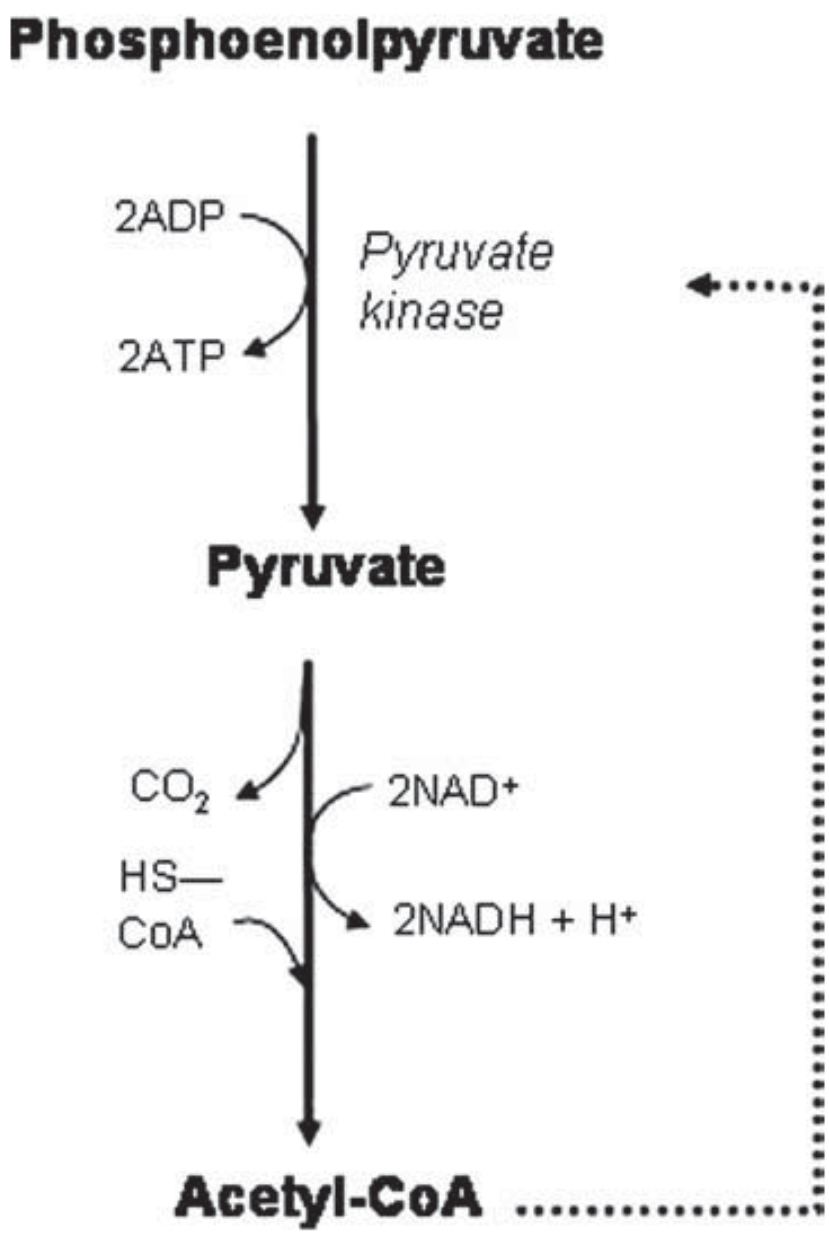


A voltage switch

- Glennan - ``Mechanisms and the nature of causation''. 1996



- Bechtel & Abrahamsen – ``Explanation; a mechanistic alternative.'' 2005



Bechtel & Abrahamsen – ``Explanation; a mechanistic alternative.'' 2005

Mechanisms in sociology

- The balance theorem
- Tipping point in an economy with network effects
- Clustering mechanisms in like/dislike networks
- …

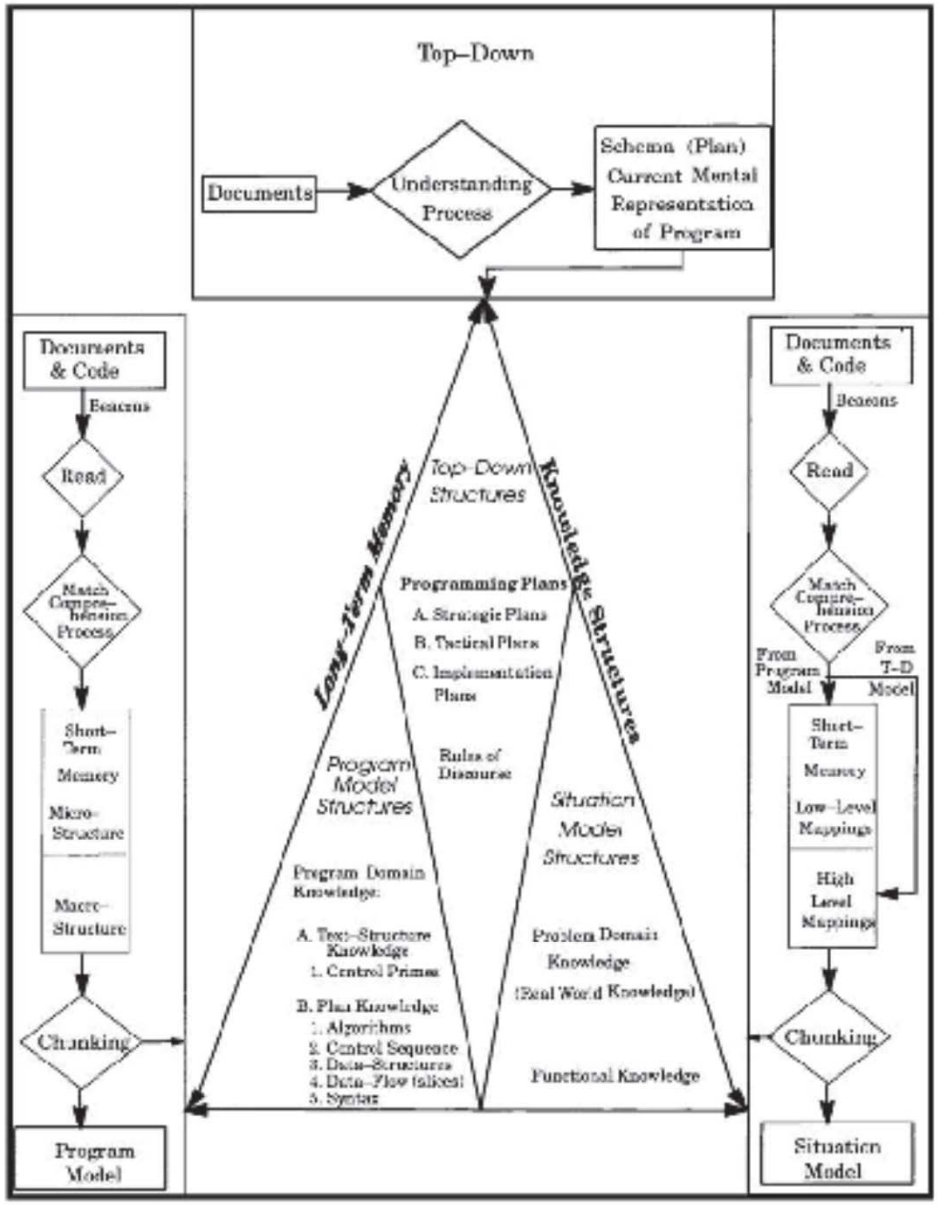


Figure 1. Diagrammatic representation of the Integrated Comprehension model from von Mayrhauser and Vans (1995a)

Cognitive mechanism of program comprehension

J. Softw. Maint., 9, 299–327 (1997)

Theory of cognitive dissonance

- *Conceptual framework*
  - *Class of systems: people interacting with the real world*
  - *Components: beliefs, intentions, values, facts, observations, conflict between facts and observations*
  - *Capabilities of people:*
    - *Change behavior*
    - *Change a value*
    - *Change intention*
    - *Deny observation*
    - *Deny fact*

- Conceptual framework provides a way to describe phenomena (a frame)
- but does not itself provide generalizations about phenomena
  - They cannot be falsified!
  - They can be useful, i.e. contribute to a purpose

Other examples

- In all of these examples we recognize in each case/population element an architecture consisting of components that have capabilities to interact.
  - *Theory of electromagnetism*
  - *The Balance theorem in social networks*
  - *Technology Acceptance Model*
  - *Theories X, Y, Z, and W of (project) management*

More examples

- Examples
  - *A software prototype in a simulated context*
  - *Software engineering projects in the real world*
  - *Information systems in organizations*
  - *Risk assessment methods for IT security*
  - *Business networks*
- Which architecture to recognize in a case? That depends on which generalizations we would like to be able to draw.

Advantages of architectural structures

Architectures can be used to decompose complex problems into simpler problems

- Study one component at a time;
- Study a system architecture while abstracting from internal structure of components

This in turn can be used for

- **Explanation, diagnosis, debugging**: Trace system-level phenomena to component properties.
- **Prediction, design**: Explore the effects of putting different components together.
- **Generalization**: Reason about similarity of cases or populations.

Statistical structures

- The world is structured as a population.
  - A **population** is set of all objects that satisfy a predicate, called the population predicate.
  - Internal structure of population elements is ignored.
  - A **(random) variable** is an observable property of population elements
  - A **probability distribution** of a variable X is a mathematical function that summarizes the probability of randomly selecting a value of X from the population

- (``Random'' means: No systematic selection proces. Internal structure of the selection process does not provide information about the outcome of the selection.)

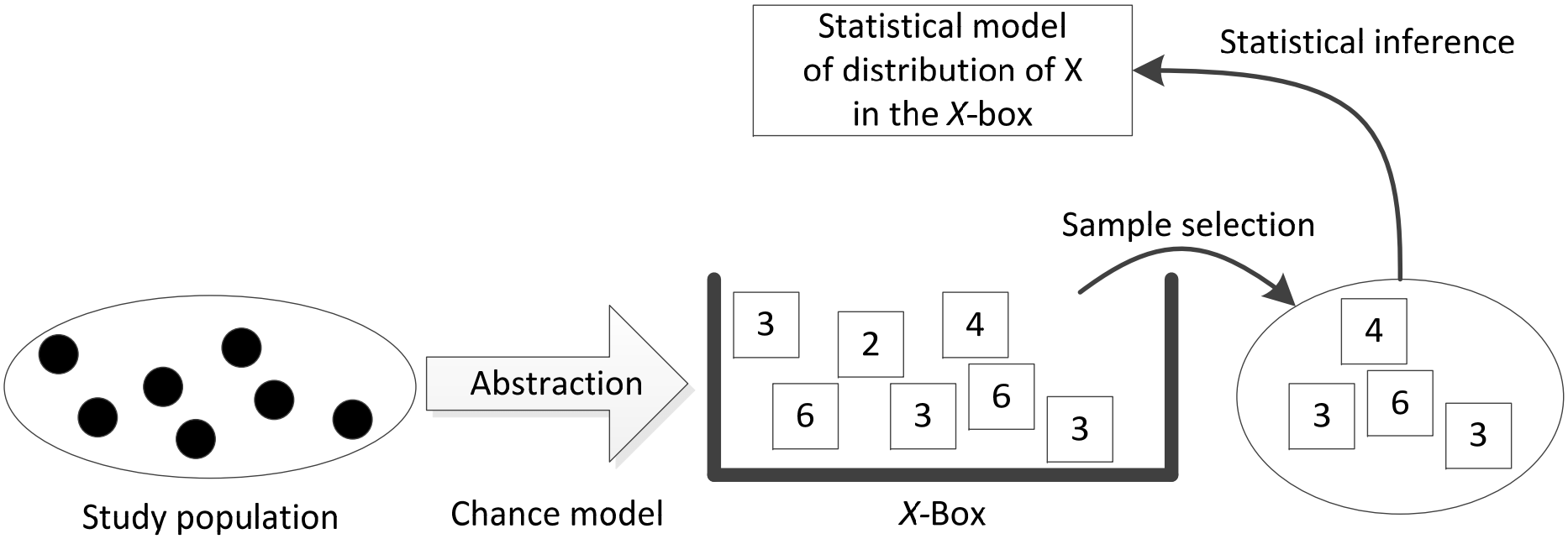
Population

- **Population** is set of all objects that satisfy a predicate, called the *population predicate*.
  - *All software prototypes in a simulated context, similar to a particular prototype & context*
  - *All agile software engineering projects in the real world*
  - *All ERP systems in large organizations*
  - *All risk assessment methods for IT security*
  - *All business networks*

- The population predicate is a **similarity predicate**
  - When defining a population predicate, we **do** often consider internal structure of population elements.

Variables

- A **(random) variable** is an observable property of population elements
- **X-box** is the multiset of values of X on a population
  - *Weight of people, effort of SE projects, numbers on football shirts.*



Statistical reasoning is about boxes

- **XY-box** is the multiset of pairs of values of (X, Y) on a population, etc.
  - *Execution time of implementations, accuracy of output, effort spent on a project, duration of a project, (effort, duration) of a project, etc.*

- In a population,
  - Each variable has properties of variation (average, variance)
  - And sets of two or more variables have relationships of covariation

Probability distributions

- **Probability distribution** of X is a mathematical function that summarizes the probability of selecting a sample of values in a random draw from the population

Chance models

- **Chance model** of X:
  1. Definition of the meaning of numbers in the X-box
  2. Assumptions about probability distribution of X
  3. Measurement procedure
  4. Sampling procedure

Example

- T. Huynh, J. Miller, An empirical investigation into open source web applications' implementation vulnerabilities. Empir. Softw. Eng. **15**(5), 556–576 (2010)
- *Population of open source web applications*
- *Random variable ImpV indicates implementation vulnerabilities.*
- *Chance model of ImpV:*
  1. *Definition: The numbers on the tickets in the ImpV-box are proportions of implementation vulnerabilities among total number of vulnerabilities in a web application. (pages 564-565)*
  2. *Assumptions: binomial distribution. I.e. ithe proportions of implementation vulnerabilities in different web applications are independent, and the probability that a vulnerability is an implementation vulnerability, is constant across all web applications*
  3. *Measurement procedure: Counting and classifying by a person.*
  4. *Sampling procedure: Not specified. 20 applications are listed.*

Advantages of statistical structures

- Statistical structures can be used to make large-scale population properties visible
  - Even when an individual shows no regularities

This in turn can be used to

- Describe aggregate phenomena in a sample
- Generalize from a sample to a population (sample-based)
  - Estimate patterns in the population not visible at the individual level (e.g. identify needs in a population)
  - Estimate the effect of treatments in the population (prediction of policy impact)
- Useful for policy-makers

Hybrid structures

- Statistical study within a case
  1. *Case study of a large global SE project: e.g. an architectural study of roles and interactions among software engineers;*
  2. *Contains a statistical study of the population of software engineers in this project.*

- Case study of a population element
  1. *Statistical survey of a sample of projects:*
  2. *Followed by case studies of a few projects, in order to understand mechanisms responsible statistical results.*

If the studied phenomena contain people …

they too have conceptual frameworks

Sharing and interpreting a conceptual framework

- Concepts shared by people in the domain may be adopted by researchers that investigate the domain
  - *Goal, requirement, effort, etc.*
  - Adopting these concepts in the conceptual research framework may allow additional understanding
- Concepts defined by researchers may be adopted by people in the domain
  - *(software) object program structure, agile, etc.*
  - Adopting these concepts in the domain may allow definition of additional options for action
- Concepts may even make a round trip from domain to researchers to domain

Validity of a conceptual framework

- Conceptual framework cannot be true or false
  - A definition cannot be true or false
- It is a tool
  - People may be able to use it (usability)
  - … for a useful purpose (utility)
- *E.g. the different definitions of "goal", "requirement", "specification", design".*
- *``A goal is the space between two trees"*

Construct Validity

- **Construct validity** is the degree to which the application of constructs to phenomena is warranted with respect to the research goals and questions

- ``Construct validity is the degree to which a test measures what it claims, or purports, to be measuring.”

- ``Construct validity refers to the degree to which inferences can legitimately be made from the operationalizations in your study to the theoretical constructs on which those operationalizations were based.’’

Threats to construct validity

- Inadequate definition
  - No identification and classification criterion.
  - *E.g. ``agile projects'' without instruction how to recognize and count agile projects*
- Construct confounding
  - Our cases/sample may be cases/samples from more than one population. Then what is the target of generalization?
  - *Measuring the effect of an SE method in a sample of students.*
    - *Is this a sample of novice software engineers? Well educated software engineers? Captive software engineers?*

Operationalizations

- Some constructs cannot be measured
- **Operationalize** them:
  - Define them in terms of one or more indicators that *can* be measured
  - An **indicator** is a variable that can be measured
  - In software engineering, often called a *metric.*

Threats to validity of operationalizations

- **Mono-operation bias**
  - Defining only one indicator for a construct
  - *E.g. measuring maintainability by effort to repair a bug only (and ignoring effort to find a bug or to test the repair).*
- **Mono-method bias**
  - Indicator measured in only one way.
  - *E.g. measuring* effort to repair *by questionaires only*

Construct Validity, again

- **Construct validity** is the degree to which the application of constructs to phenomena is warranted with respect to the research goals and questions
- ``Construct validity is the degree to which a test measures what it claims, or purports, to be measuring.”
  - The paradox of analysis: if a vague concept is operationalized, either the operationalization is wrong or the concept is redefined
- ``Construct validity refers to the degree to which inferences can legitimately be made from the operationalizations in your study to the theoretical constructs on which those operationalizations were based.’’
  - Indicators do not *causally influence* constructs
  - Paradox of analysis: construct validity is either 0 or 100%

Outline

1. What is design science
2. Research goals and problems
3. The design and engineering cycles
4. **Theories**
   - Conceptual frameworks
   - **Generalizations**
5. Empirical cycle
6. Scientific inference

Target (scope) of generalization

- In case-based reasoning: defined by **similarity predicate**
  - *Agile projects: Projects following one of the recognized agile methods such as Scrum or XP. How crisp is this?*

- In sample-based reasoning: Defined by similarity predicate too!
  - Population predicate
    - *Generalize, from sample of study population of agile projects, to the study population*
  - Similar populations
    - *Generalize from study population of agile projects to the theoretical population of all agile projects.*

The structure of design generalizations

- (Artifact specification) X (Context assumptions) → Effects
  - *(UML) X (SE project) → (Less errors)*
  - *(Cross-functional teams) X (Global SE project) → (Less errors)*
  - *(Algorithm) X (Context of use) → (Faster execution)*
- Effects satisfy a requirement to some extent
  - Often, researchers have no clear requirements
  - Requirements come from stakeholder goals, so the relevance of a generalization depends on stakeholder goals

Discussion

- Do you recognize this?
- More examples …

Why do we need theories?

- With a conceptual framework we can:
  - **Frame** a problem or artifact: *choose which concepts to use*
  - **Analyze** conceptual problem structure: *analyze the framework*
  - **Describe** a problem or artifact
  - **Generalize** about problem or artifact
- With a generalization we can
  - **Explain** phenomena
  - **Predict** phenomena

Examples

- Theories:
  - *Theory of cognitive dissonance*
  - *Theory of electromagnetism*
  - *The Balance theorem in social networks*
  - *Theories X, Y, Z, and W of (project) management*
  - *Technology Acceptance Model*
- What can we do with these theories:
  - **Frame** a problem or artifact
  - **Analyze** conceptual problem structure
  - **Describe** a problem or artifact
  - **Generalize** about problem or artifact
  - **Explain** phenomena
  - **Predict** phenomena

Explanations

- An **explanation** is hypothesis about how a phenomenon came about.
  - **Causal explanations** explain the occurrence of an event by the occurrence of an earlier event
  - **Architectural explanations** explain the existence of a causal relationship by the mechanisms that produced it
  - **Rational explanation**s explain the behavior of actors by their goals.
- *E.g. why is the light on?*

Causal explanations

- Causal explanations say that an earlier event **made a difference** to a current event.
  - *Explanation of lower programnming effort: Programming effort is lower **because** we switched to UML*
    - *The earlier switch to UML resulted in the current reduction of programming effort*
    - *If we had not switched to UML earlier, our current programming effort would have been higher, other things being equal.*

- Causality is difference-making.
  - It is unobservable.
  - It may be nondeterministic

Architectural explanations

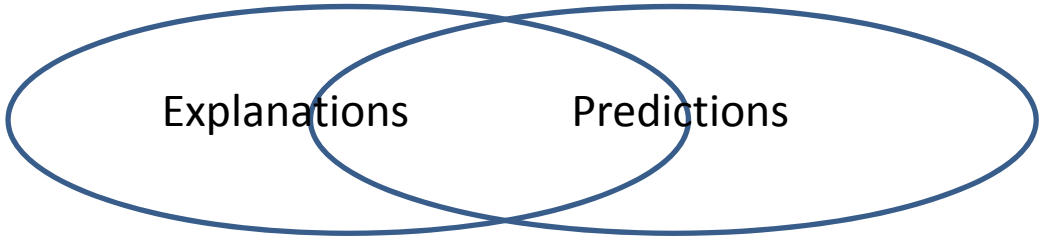
- **Architectural explanations** explain the existence of a causal relationship by the mechanisms that produced it.
  - *Explanation why UML leads to lower programming effort:*
    - *UML models resemble the domain more than other kinds of models;* → Capability of UML
    - *they are easier to understand for software engineers;* → Cognitive mechanism
    - *They allow seamless transition from domain models to software models.* → Mechanism in SW development
- A **mechanism** is the collection of interactions among components (entities, actors) that produces a response from a stimulus.
  - Mechanisms may be social, psychological, physical, technical, …
  - Mechanisms can be observed; capabilities are unobservable
  - Mechanisms may be nondeterministic

Rational explanations

- **Rational explanation**s explain the behavior of actors by their goals.
  - *Why do we have lower programming effort? Because we bid for fixed price contracts. We need to be sure that our software engineers work as efficiently as possible*
- Architectural explanation for social systems that include rational actors
  - We assume rationality in users, programmers, attackers, fraudsters, thiefs, politicians, … but their goals may not be our goals.

Predictions

- Explanations say how the present is an outcome of the past.
- Predictions look at the present and estimate what the future will be.

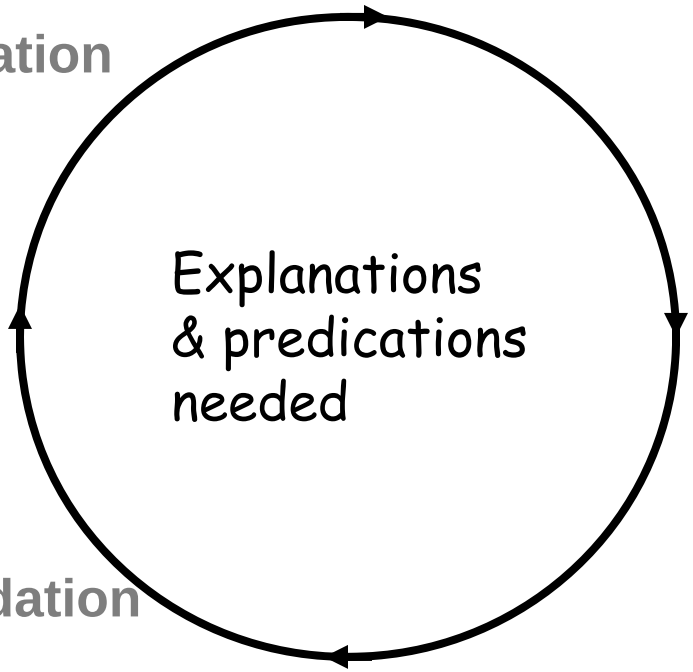


- Some explanations are too incomplete to be used as predictions
  - *Explanations of the outcome of a football match*
- Some predictions are descriptive, and do not provide an explanation
  - *In CMM 3 organizations developing embedded software, defect removal effectiveness is 98%.*
- Some explanations can be used for prediction too
  - *Most examples of explanations given so far.*

The use of theories in design

Treatment implementation

Implementation evaluation = Problem investigation



- Stakeholders? Goals?
- Conceptual problem framework?
- Phenomena? **Explanations?**
- **Effects?** Contribution to Goals?

Treatment validation

- **Context & Artifact → Effects?**
- Effects satisfy Requirements?
- Trade-offs for different artifacts?
- Sensitivity for different Contexts?

Treatment design

- Specify requirements!
- Requirements contribute to goals?
- Available treatments?
- Design new ones!

Usability of design theories

- When is a design theory

  (Context assumptions) X (Artifact design) → Effects

  usable by a practitioner?

  1. He/she is capable to recognize Context Assumptions
  2. and to acquire/build Artifact,
  3. effects will indeed occur, and
  4. will contribute to stakeholder goals/satisfy requirements

- Practitioner has to asses the risk that each of these fails

Example

- *(UML) X (any SE project) → lower programming effort*

- *You are a practitioner.*
  1. *Can you recognize Context Assumptions?*
  2. *Can you acquire UML?*
  3. *Are you sure the effects will occur?*
  4. *Do they contribute to stakeholder goals?*

Discussion

- Gregor's view of theories.
  - My book page 103.

Outline

1. What is design science
2. Research goals and problems
3. The design and engineering cycles
4. Theories
   - Conceptual frameworks
   - Generalizations
5. **Empirical cycle**
6. Scientific inference

Checklist for the empirical cycle: context

1. Improvement goal?
2. Knowledge goal?
3. Current knowledge?

Design cycle
(slide 22)

Empirical cycle

4. …

….

16. …

17. Contribution to knowledge goal?
18. Contribution to improvement goal?

Designing something useful

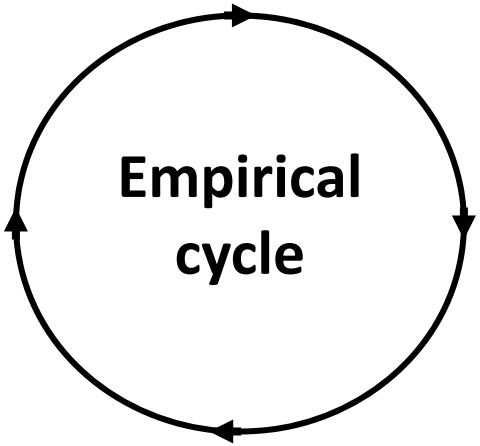
Answering a knowledge question

Data analysis

12. Data?
13. Observations?
14. Explanations?
15. Generalizations?
16. Answers?

Research execution

11. What happened?



Research problem analysis

4. Conceptual framework?
5. Research questions?
6. Population?

Design validation

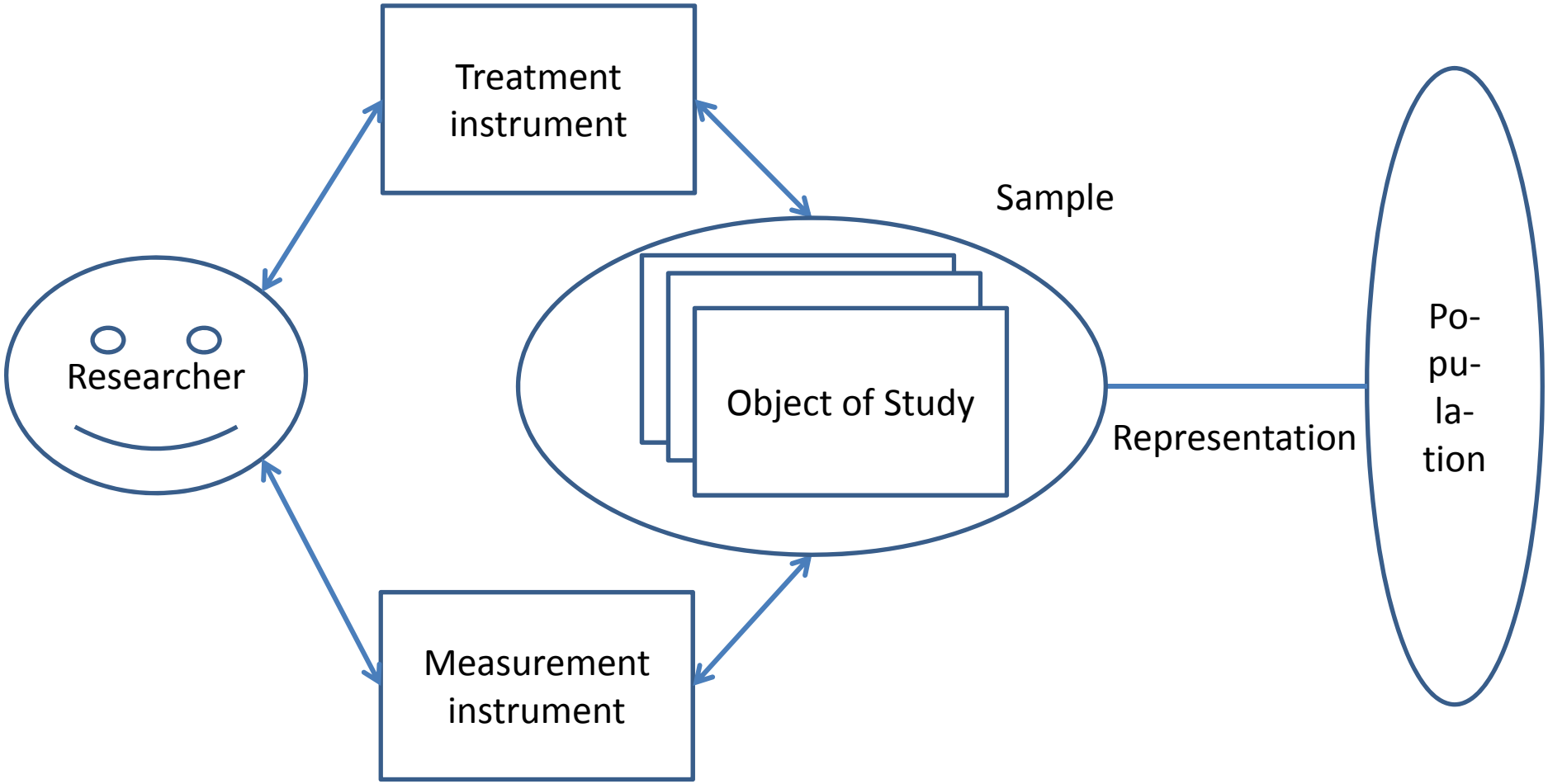
7. Object of study justification?
8. Treatment specification justification?
9. Measurement specification justification?
10. Inference justification?

Research & inference design

7. Object of study?
8. Treatment specification?
9. Measurement specification?
10. Inference?

Research setup

Research setup



- In sample-based studies, ``representation'' is achieved random sampling
  - A random sample may be **dissimilar** to any other sample from the same population.
  - No analogic reasoning from random sample to population
  - Randomness guarantees long-run stability of statistical properties
- In case-based research, OoS's are studied sequentially
- In observational studies there is no treatment
  - But all instruments interact with the OoS's.
- Which inferences you can do, depends on the research setup

Examples

- What is the population, the sample of objects of study, the measurements, and the treatments (if any)?
  - *Experiment with students to test a new SE technique to see if it leads to less programming effort*
  - *Survey of the use of UML in Brazil*
  - *Case study of IS implementation failure*
  - *Experimental use of a new effort estimation technique in a pilot project to see if it is more accurate*

Kinds of empirical research methods

	Observational study (no treatment)	Experimental study (treatment)
Case-based: investigate single cases, look at architecture and mechanisms	Observational case study	• Expert opinion • Case-based mechanism experiments • Technical action research
Sample-based: investigate samples drawn from a population, look at averages and variation	Survey	• Statistical difference-making experiment

Example observational case study

An Empirical Study of the Complex Relationships between Requirements Engineering Processes and Other Processes that Lead to Payoffs in Productivity, Quality, and Risk Management

Daniela Damian and James Chisan
TSE 32(7) July 2006

Research design

- ***Research context***

1. *Improvement goal: None. Curiosity-driven research.*
2. *Knowledge goal: Fill gap between claims about RE and RE practice*
3. *Current knowledge: A few published surveys, no case studies.*

Research design

Value-word. Avoid

- ***Research problem analysis***

4. *Conceptual framework: definitions of requirements, productivity, quality, risk*
5. *Research questions:*
   - *How do improvements in RE processes relate to improvements in productivity, quality, and risk management?*
     1. *How do improvements in the RE practice impact the early stages of development?*
     2. *How do they impact the downstream development stages?*
     3. *Which components of the RE process were more significant in contributing to this impact?*
     4. *How could the interaction between REP and other processes have contributed to these results?*
6. *Population: RE processes in software development organizations*

Research design

- ***Research & inference design***

7. *Object of study: Introduction of RE at the Australian Center for Unisys Software*
8. *Treatment specification: No treatment. Observational research.*
9. *Measurement specification: Questionnaires, interviews, document inspection.*
10. *Inference:* after the break!

Example Technical Action Research (TAR) study

Morali, A. and Wieringa, R.J. (2010) *Risk-Based Confidentiality Requirements Specification for Outsourced IT Systems.* In: *Proceedings of the 18th IEEE International Requirements Engineering Conference* (RE 2010), 27 Sept - 1 Oct 2010, Sydney, Australia. IEEE Computer Society.

What is TAR?

- Using an experimental technique
  - to help a client and → Helping a client
  - to learn about its effects in practice. → Experimenting with a technique

TAR methodology

TAR

Researcher's design cycle:
Design a new technique

Researcher's empirical cycle:
Validate a new technique

Client's engineering cycle:
Solve a problem

TAR methodology

TAR

Researcher's design cycle:
- Investigate problem
- Design treatment
- Validate treatment

Researcher's empirical cycle:
- Analyze research problem
- Design research setup and inferences
- Validate
- Execute
- Analyse results

Client's engineering cycle:
- Investigate client problem
- Design treatment
- Validate treatment
- Implement
- Evaluate

TAR methodology

TAR

Researcher's design cycle:

- Investigate problem
- Design treatment
- Validate treatment

Researcher's empirical cycle:

- Analyze research problem
  - Effects?
  - Satisfy requirements?
  - Comparison?
  - Generalizable?
- Design research setup and inferences
  - Acquire a case
- Validate
  - Internal validity?
  - External validity?
- Execute
  - Do the client cycle
- Analyse results
  - Answer questions.

Client's engineering cycle:

- Investigate client problem
  - Stakeholders, goals?
  - Phenomena?
  - Causes, effects?
- Design treatment
  - Make a plan.
- Validate treatment
  - Check with client.
- Implement
  - Do it.
- Evaluate
  - Client satisfied?

Research Design

Research context

1. *Knowledge goal*
   - *To validate a newly developed confidentiality risk assessment method*
2. *Improvement goal*
   - *To develop a confidentiality risk assessment method when IT is outsourced.*
3. *Current knowledge*
   - *The method has been tested on small problems in the lab*
   - *The method has been inspected by stakeholders of a client company*

Research Design

Research problem

4. *Conceptual framework*
   - *Concepts from outsourcing and security risk assessment*
5. *Knowledge questions*
   - *Q1 Is the technique usable?*
   - *Q2 Is it less subjective than other available techniques?*
   - *Q3 Does use of the technique improve the client's understanding of confidentiality risks, better than competing techniques do?*
   - *Q4 Is the effectiveness of the techniques dependent on its context of use? If so, how?*
6. *Population*
   - *The set of all ERP outsourcing relations where the service consumer must satisfy confidentiality requirements checked by auditors and depends on the outsourcing service provider to satisfy these requirements.*

Research design

Research & inference design

7. *Object of study*
   - *large multinational industrial company with a significant corporate security staff keen on improving their procedures.*
8. *Treatment design*
   - *The RA method designed by the researcher was customized for the company.*
   - *Some software tools were used.*
   - *A schedule of interactions was agreed with the company.*

Research design

Research & inference design, continued

9. Measurement design

- *Usability measured by effort (time) to use;*
- *Subjectivity measured by number of questions in the method that require personal judgment;*
- *Client's understanding measured by debriefing interview*

– *Data sources: Architects, Chief security officer, primary documents*

– *Instruments: Diary of researcher, Interviews with key stakeholders*

– *Measurement plan: Agreements about what data could be collected, and how*

- *Inference design:* <u>After the break!</u>

Outline

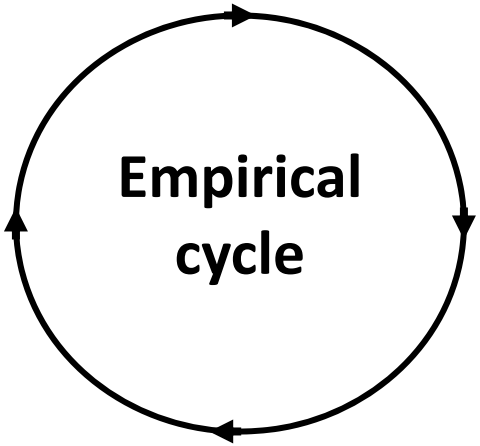
1. What is design science
2. Research goals and problems
3. The design and engineering cycles
4. Theories
   - Conceptual frameworks
   - Generalizations
5. Empirical cycle
6. **Scientific inference**

Data analysis

12. Data?
13. Observations?
14. Explanations?
15. Generalizations?
16. Answers?

Research execution

11. What happened?



Research problem analysis

4. Conceptual framework?
5. Research questions?
6. Population?

Design validation

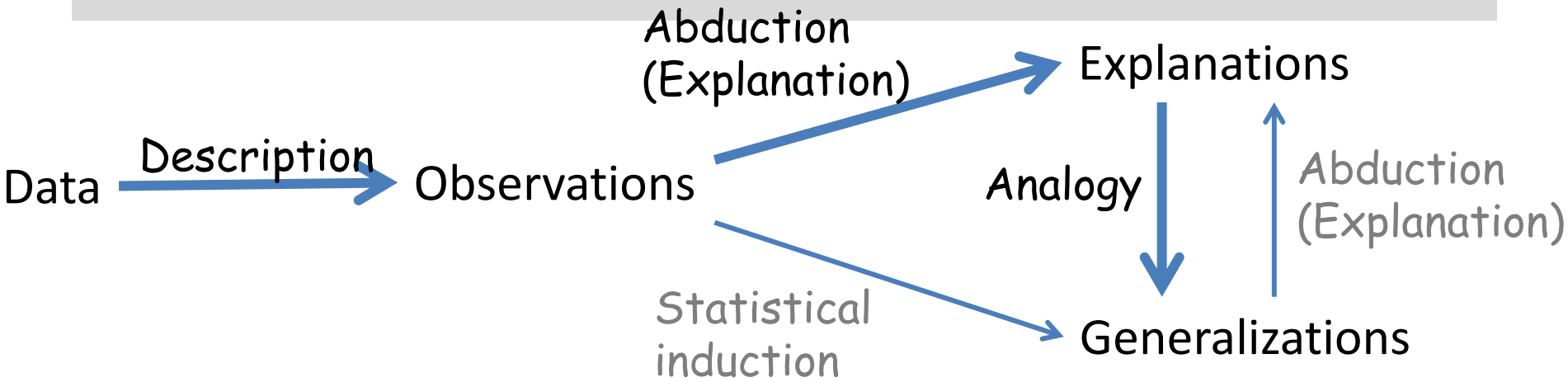
7. Object of study justification?
8. Treatment specification justification?
9. Measurement specification justification?
10. Inference justification?

Research & inference design

7. Object of study?
8. Treatment specification?
9. Measurement specification?
10. Inference?

Research setup

Case-based inference design



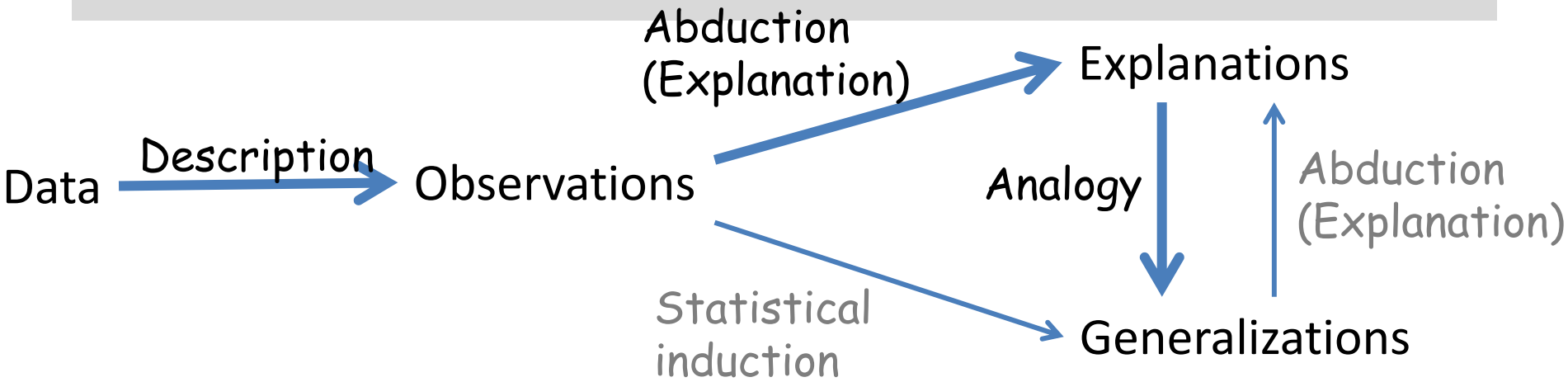
Case-based research

1. Describe
2. Explain in terms of architecture & mechanisms
3. Generalize by architectural analogy

Sample-based research

1. Describe
2. Statistically infer study population properties
3. Explain in terms of variables & causes
4. Generalize to theoretical population by architectural analogy

Case-based inference design



Case-based research

1. Describe
2. Explain in terms of architecture & mechanisms
3. Generalize by architectural analogy

Sample-based research

1. Describe
2. Statistically infer study population properties
3. Explain in terms of variables & causes
4. Generalize to theoretical population by architectural analogy

Descriptive validity

Internal validity

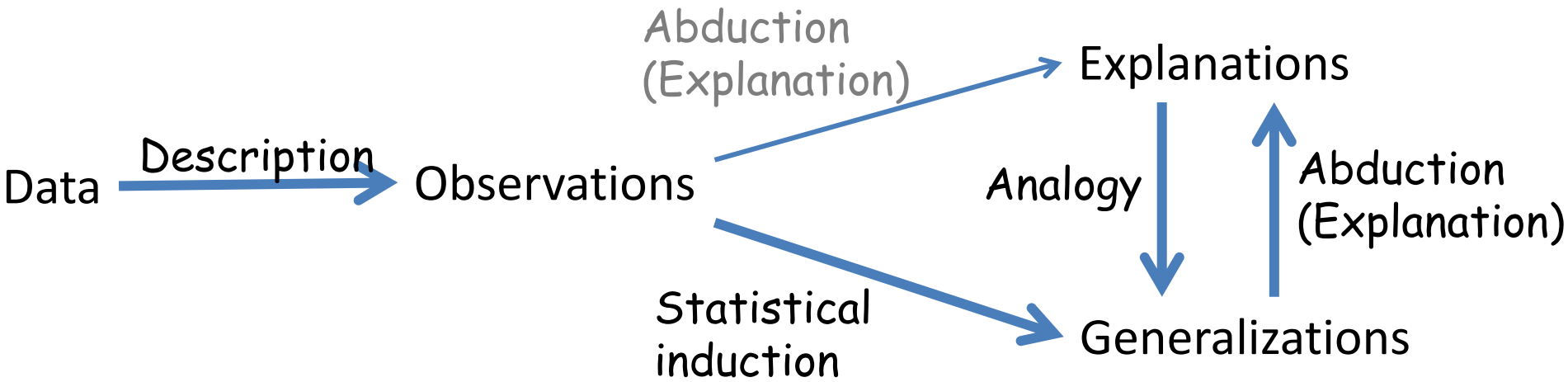
External validity

Validity

- Validity is the degree of support for your inferences
  - It is not the degree of truth of your conclusions

- Validity discussion: discuss all of the possible ways in which your inferences could be wrong
  - Legal reasoning: your opponent will try to undercut all your arguments.

Sample—based inference design



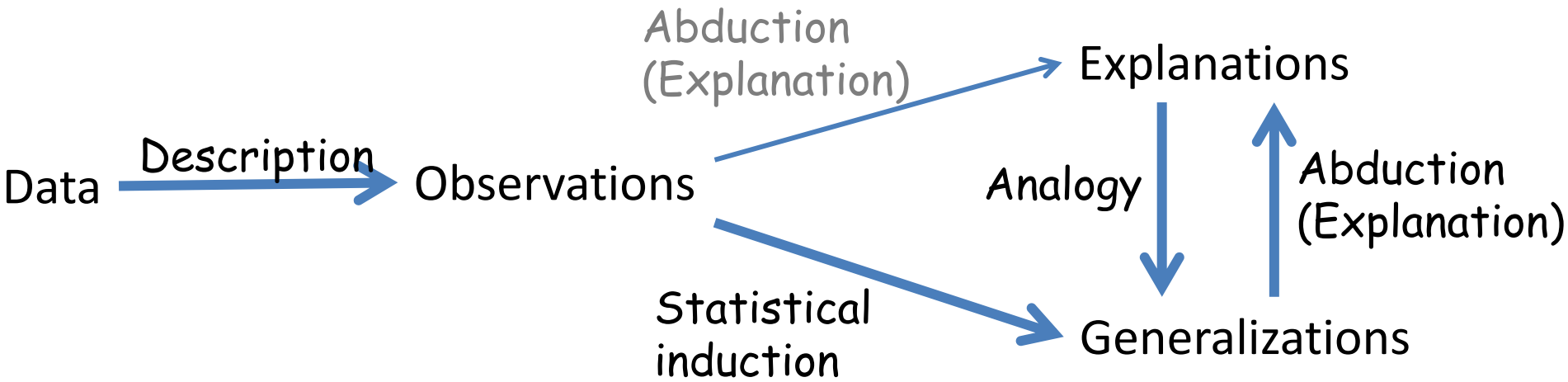
Case-based research

1. Describe
2. Explain in terms of architecture & mechanisms
3. Generalize by architectural analogy

Sample-based research

1. Describe
2. Statistically infer study population properties
3. Explain in terms of causes, mechanisms, reasons
4. Generalize to theoretical population by architectural analogy

Sample—based inference design



Case-based research

1. Describe
2. Explain in terms of architecture & mechanisms
3. Generalize by architectural analogy

Descriptive validity

Statistical conclusion validity

Internal validity

External validity

Sample-based research

1. Describe
2. Statistically infer study population properties
3. Explain in terms of causes, mechanisms, reasons
4. Generalize to theoretical population by architectural analogy

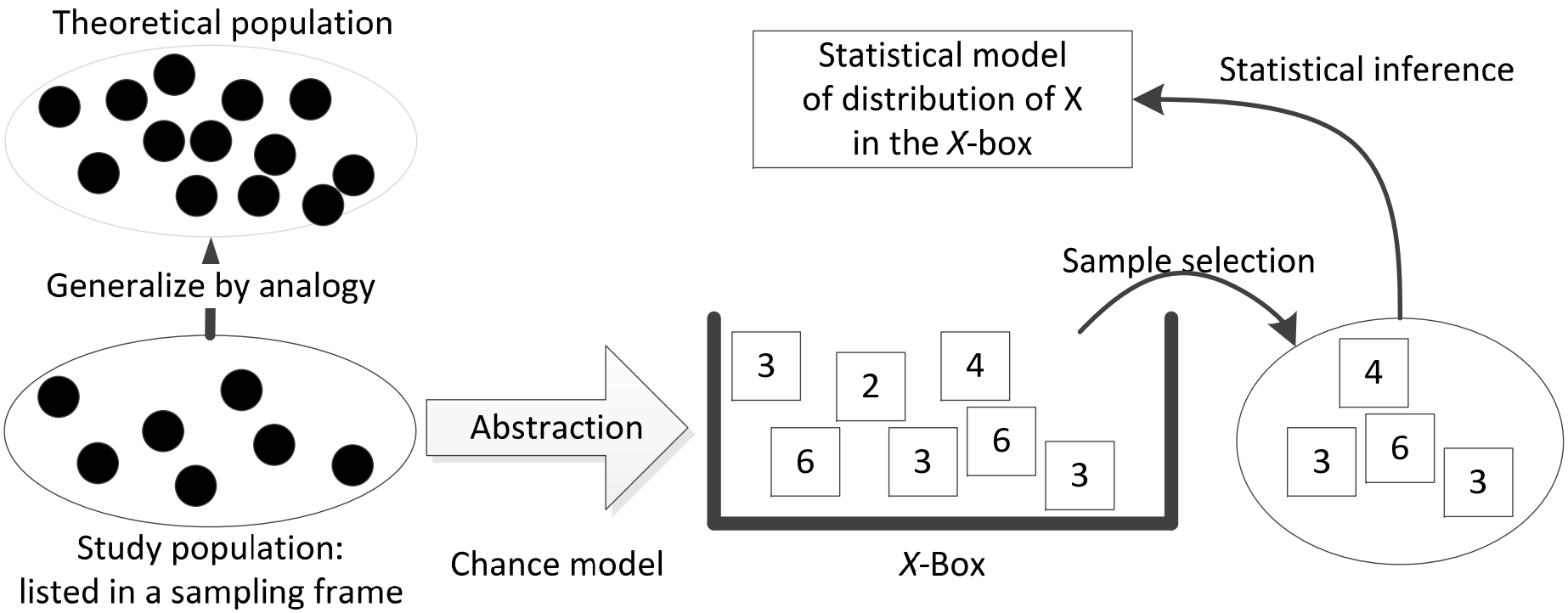
Descriptive inference design

- **Descriptive inference**
  i. Data preparation
     - Deal with missing data, outliers, perform transformations, etc.
  ii. Data interpretation
     - Conceptual analysis of text, images etc., content analysis, grounded theory, etc.
  iii. Summaries
     - Textual, numerical, graphical, qualitative.
- **Descriptive validity**
  - Triangulation (multiple sources, multiple interpretation methods, mutliple coders)
  - Member checking: check with subjects
  - Peer debriefing: check with peers

Statistical inference design

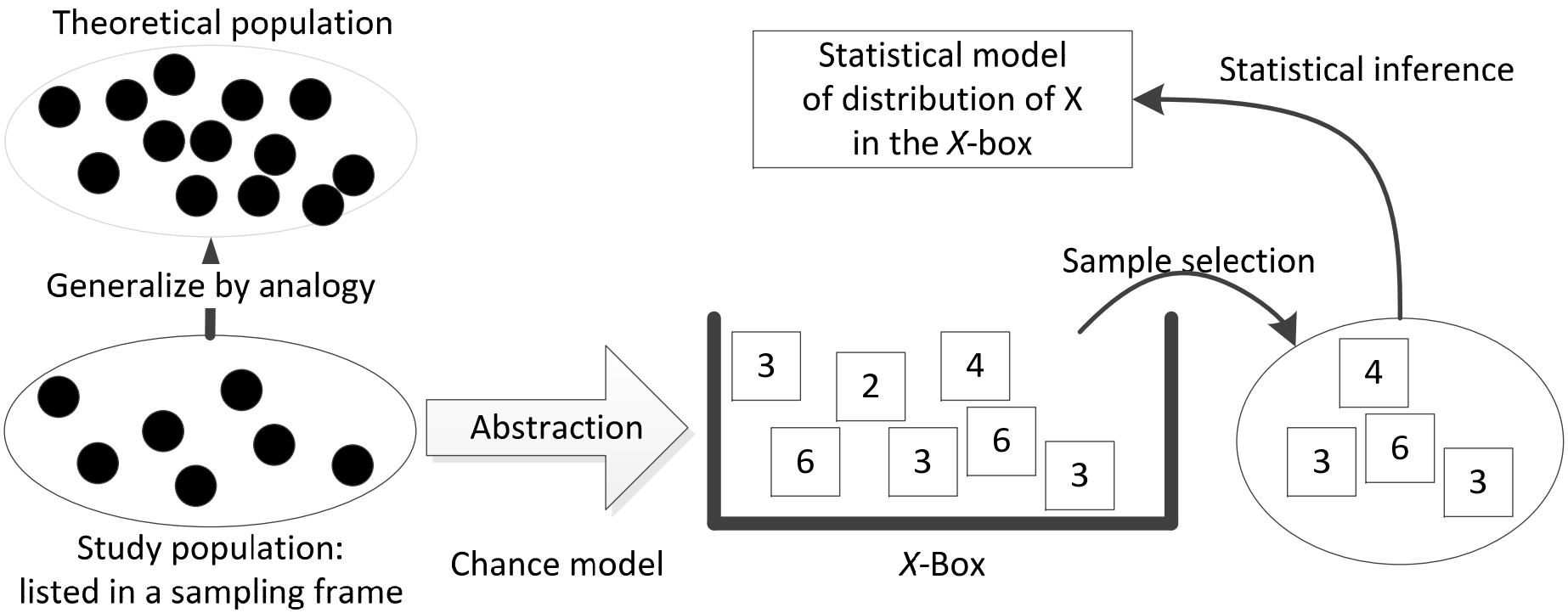
- **Statistical inference**
  - There can be no statistical inference from a sample of one or more case studies
  - To use statistical inference in sample-based designs, check the assumptions of the statistical inference techniques.

Targets of generalization



- The study population is a subset of the theoretical population.
- What is the basis for generalizing to the theoretical population?
- Can we generalize numerical average and variance to the theoretical population?

Targets of generalization



- Case studies are about the analogies among the bullets

Abductive inference design

- **Abduction**
  - What possible explanations of case phenomena can you foresee?
    - Causes?
    - Mechanisms?
    - Reasons?
- **Internal validity**
  - What justifications of these explanations can you foresee?
  - Prior knowledge (common sense, scientific theories)

Analogic inference design

- **Generalization by analogy**
  - What is the scope of your explanation?
  - Similarity predicate of this class of cases? Shared architecture
- **Analytical induction**
  - Select a series of cases that you expect to confirm/disconfirm your generalization. If expectation is violated,
  - Redefine the similarity predicate to define away the problem or
  - Update the generalization to apply to all cases so far.
- **External validity**
  - Under which assumptions is the generalization justified?

Steps in sample-based research

- Define theoretical population (similarity predicate)
- Define study population (list of elements that serves as sampling frame)
- Define chance model
- Select sample
- Do measurements

1. Describe sample demographics & measurements
2. Statistically infer study population properties (~~test a hypothesis about~~, estimate confidence interval of, population parameter)
3. Explain in terms of causes, and explain causes in terms of mechanisms
4. Generalize to theoretical population by architectural analogy

Steps in case-based research

- Define theoretical population (similarity predicate)
- Select next case (analytical induction strategy)
- Do measurements

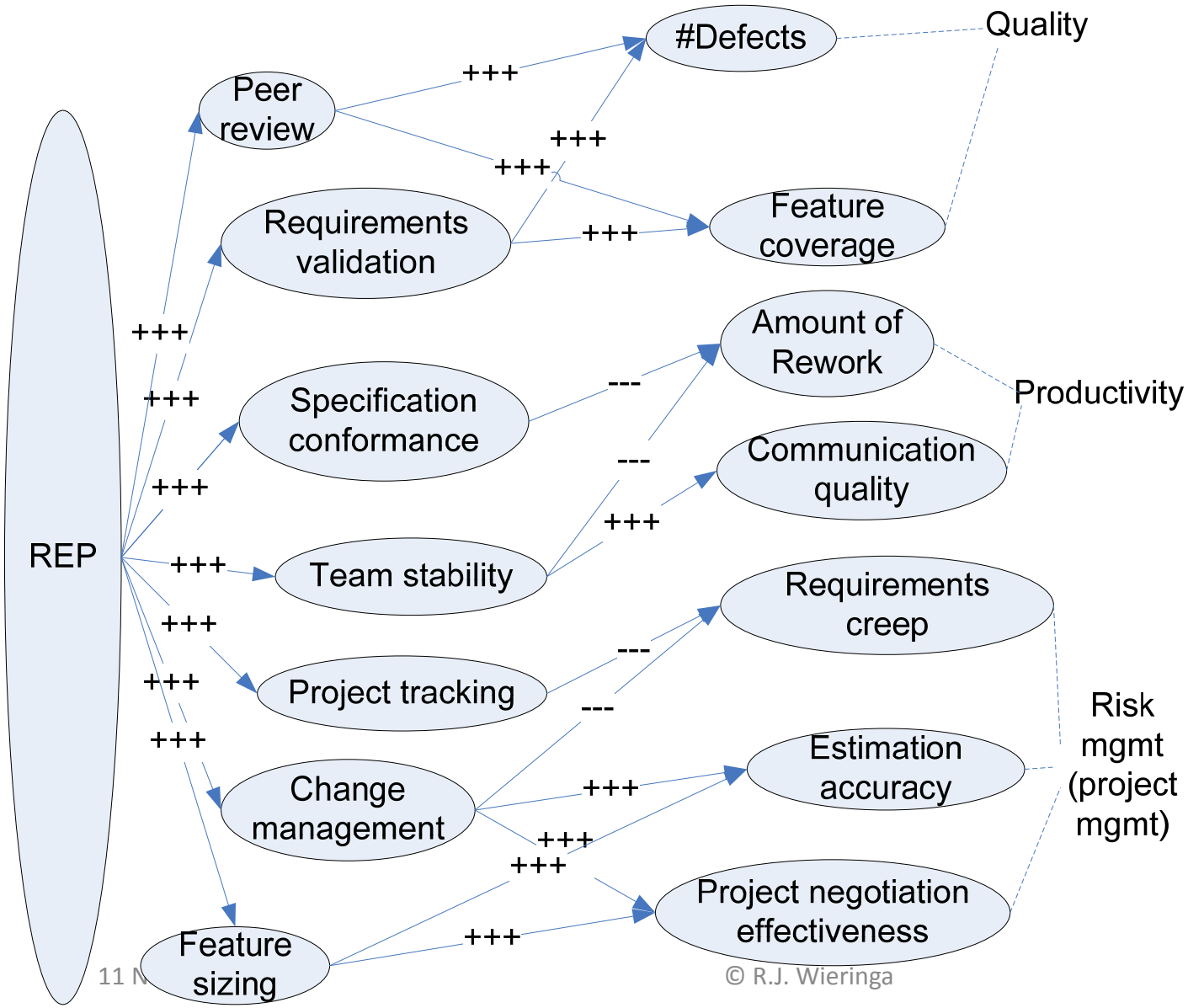
1. Describe case architecture & measurements
2. Explain in terms of architecture & mechanisms
3. Generalize by architectural analogy

Inferences in example observational case study

An Empirical Study of the Complex Relationships between Requirements Engineering Processes and Other Processes that Lead to Payoffs in Productivity, Quality, and Risk Management

Daniela Damian and James Chisan
TSE 32(7) July 2006

Damian & Chisan. RE introduction in a development company. TSE July 2006.

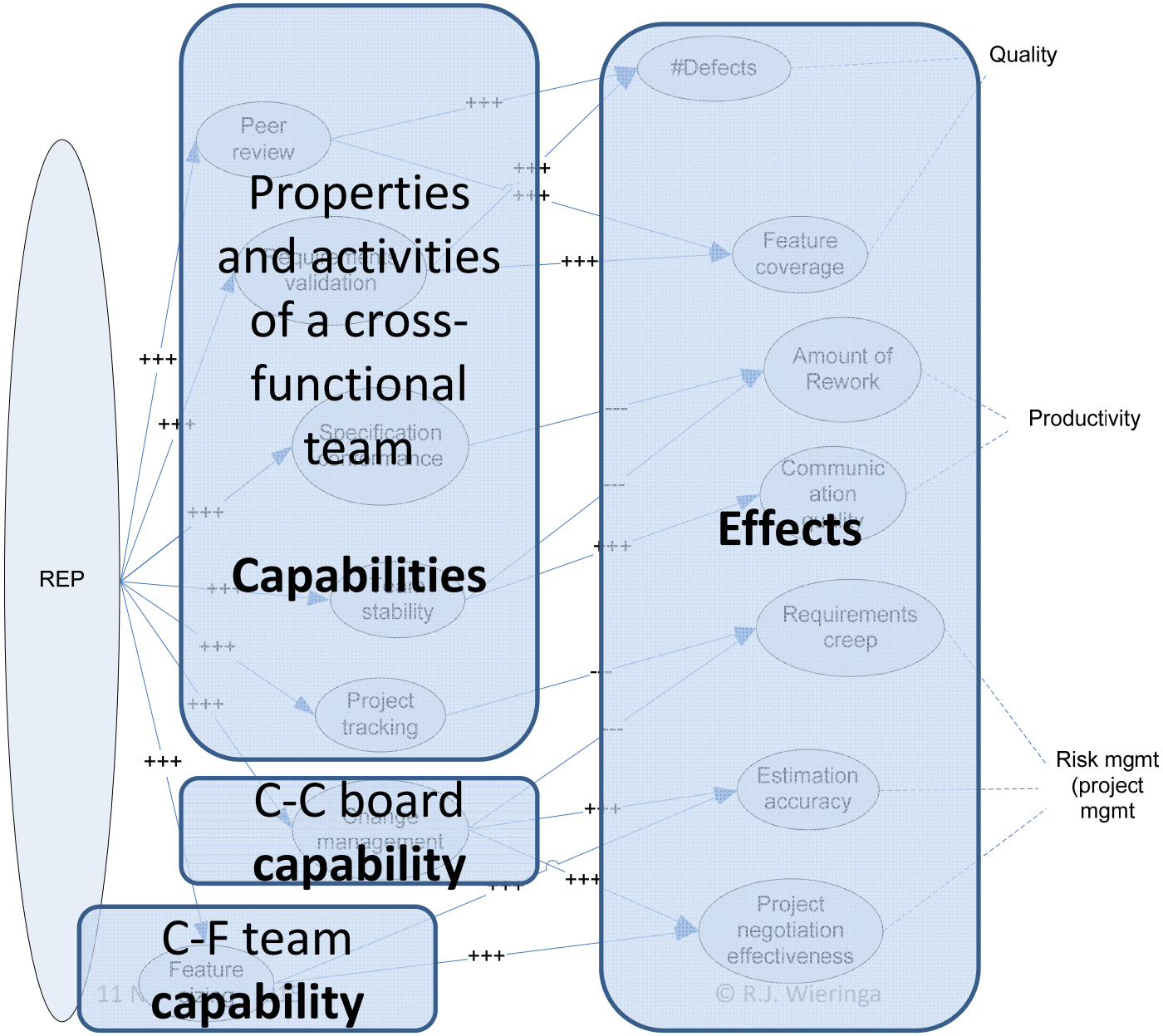


A causal model presented in the paper.

Practioners' opinions!

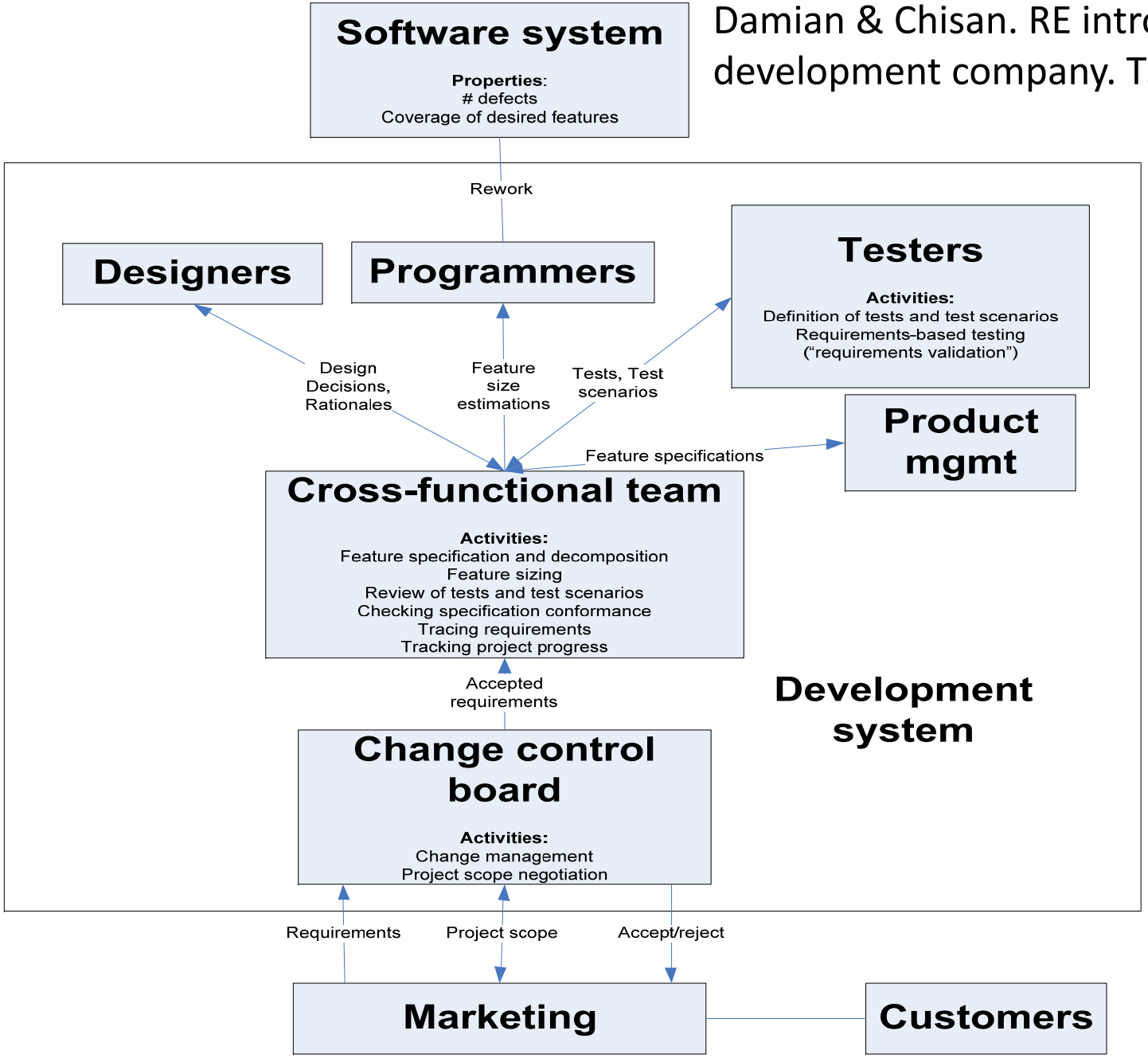
Researcher-observed practitioner-perceived correlations

Damian & Chisan. RE introduction in a development company. TSE July 2006.



The variables characterize the way of working of a cross-functional team, and its effects

Damian & Chisan. RE introduction in a development company. TSE July 2006.



An architectural model constructed from the paper.

Mechanisms that explain improved productivity, quality, risk mgmt:

C-F team

- *improves shared feature understanding,*
- *reduces rework*
-

C-C board

- *prevents requirement creep*

Which inferences can we now support?

- ***Abduction:***
  - *It is plausible that this architecture promotes these mechanisms, which contributed to the effect*
  - *Other mechanisms in this case that may have contributed to these effects:*
    - *New management may have impacted the effect variables as well*
    - *Other processes were changed too*
    - *The product was mature*
- ***Generalization:***
  - *About 150 members, small teams, tech, project, product mgrs*
  - *Ineffective customer communication*

- Need analytical induction to say more

Example Technical Action Research (TAR) study

Morali, A. and Wieringa, R.J. (2010) *Risk-Based Confidentiality Requirements Specification for Outsourced IT Systems.* In: *Proceedings of the 18th IEEE International Requirements Engineering Conference* (RE 2010), 27 Sept - 1 Oct 2010, Sydney, Australia. IEEE Computer Society.

Inference as designed

- ***Descriptive inference***
  - *Diagrams and other work products of applying the method are to be presented*
- ***Abduction***
  - *The production of the outcome (a risk assessment) is explained by the application of the method.*
  - *Treatment (the method) was applied correctly: documenting the method in advance and continuously referring to it during use*
  - *No other mechanism that could account for outcome*
  - *Other explanations: competence of the researcher who applied the method*
- ***Generalization***
  - *Scope: Large organizations; Outsourcing; Significant security department; Obligation to show that company is in control of their information assets.*

Inferences actually done

- ***Descriptions of the sample of OoS***
  - *Some of the produced diagrams are reported*
  - *A few notes from primary documents and interviews are reported.*
- ***Explanations***
  - *All products of applying the method were produced as prescribed by the method; the reason is that the researcher is the author of the method and understands exactly what needs to be done.*
- ***Generalizations***
  - *Assumptions about context that make the scope more specific:*
    - *Industrial organization, very cost-sensitive environment (they often take confidentiality risks in order to save money)*
    - *Other users would need proper tool support*

Validity

- Research setup must be valid wrt planned inferences
- Inferences actually done must be valid wrt research setup

Validity of research setup

- Inference support
  - What inferences can be supported by this research setup?
- Repeatability
  - Can other researchers repeat this research?
- Ethics
  - Does the setup satisfy ethical norms? (informed consent, no harm, fairness, confidentiality etc.)

Validity of inferences

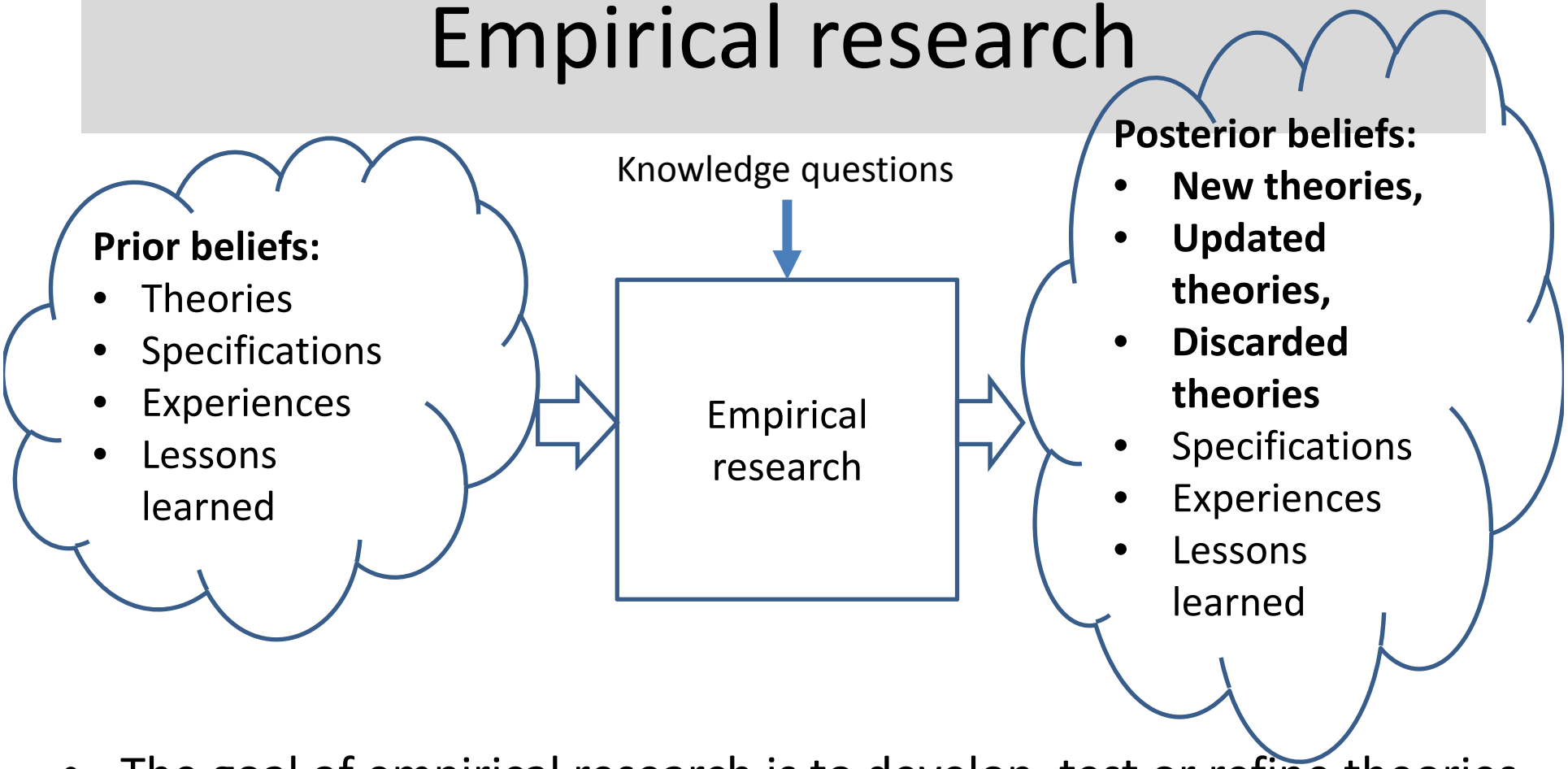
- See chapters 12 (descriptive inference), 14 (abductive inference), 15 (analogic inference)
- Summarized in checklist B, http://wwwhome.cs.utwente.nl/~roelw/Springer-Appendix-B.pdf.

Outline

1. What is design science
2. Research goals and problems
3. The design and engineering cycles
4. Theories
   - Conceptual frameworks
   - Generalizations
5. Empirical cycle
6. Scientific inference

The rules of the game

Empirical research



- The goal of empirical research is to develop, test or refine theories
- We never start empty-handed

Rules of the empirical research game

- **Rule of posterior knowledge**: Beliefs created by the research are present *after* execution of the research, and are absent before executing the research.
  - *E.g. formulating hypotheses after the fact, and reporting that you had them before*

- **Rule of prior ignorance**: Any beliefs present *before* doing the research may influence the outcome of the research.
  - *E.g. believing that the tested technique is better than its alternatives*

- If you violate these rules, you cheat yourself and your readers

Assignment

- Make a design of your research
  - Top level design cycle
  - An example empirical research design within this design cycle.
- Summarize on a flip-over

- Wieringa, R.J. and Daneva, M. (2015) *Six strategies for generalizing software engineering theories.* Science of computer programming, 101. pp. 136-152.

- Wieringa, R.J. (2014) *Design science methodology for information systems and software engineering.* Springer Verlag

- Wieringa, R.J. (2014) *Empirical research methods for technology validation: Scaling up to practice.* Journal of systems and software, 95. pp. 19-31.

- Wieringa, R.J. and Morali, A. (2012) *Technical Action Research as a Validation Method in Information Systems Design Science.* In: *Design Science Research in Information Systems. Advances in Theory and Practice 7th International Conference*, DESRIST 2012, 14-15 May 2012, Las Vegas, USA. pp. 220-238. Lecture Notes in Computer Science 7286. Springer.

- Wieringa, R.J. (2010) *Relevance and problem choice in design science.* In: *Global Perspectives on Design Science Research (DESRIST). 5th International Conference*, 4-5 June, 2010, St. Gallen. pp. 61-76. Lecture Notes in Computer Science 6105. Springer.

- Wieringa, R.J. (2009) *Design Science as Nested Problem Solving.* In*: Proceedings of the 4th International Conference on Design Science Research in Information Systems and Technology*, Philadelphia. pp. 1-12. ACM.